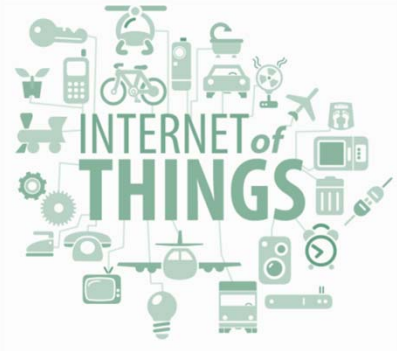
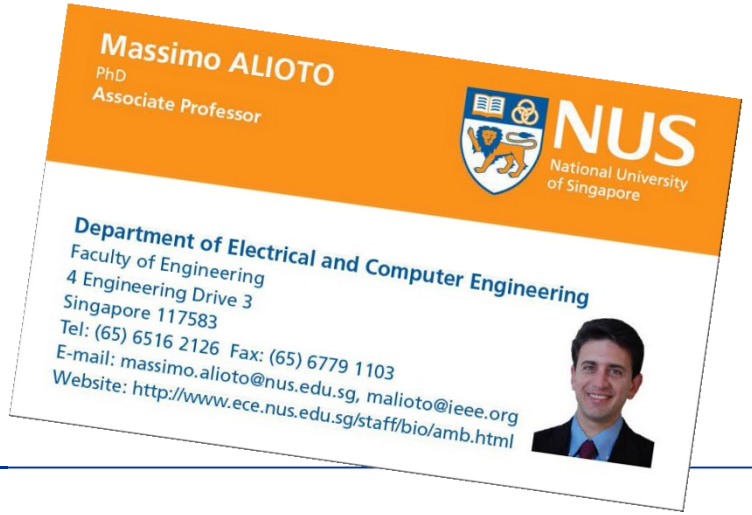
Ultra-Low Power Design Approaches for IoT



Massimo Alioto

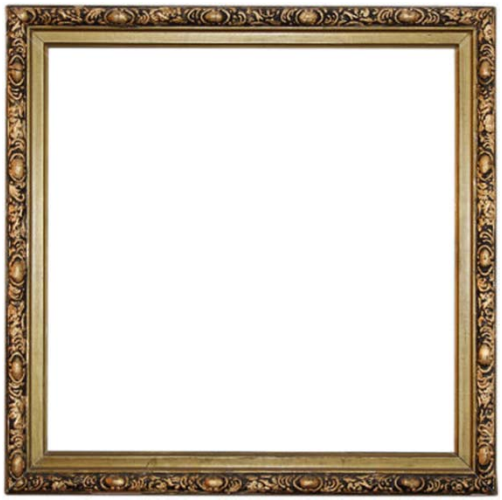
National University of Singapore (NUS)
ECE Department
Green IC group



Outline

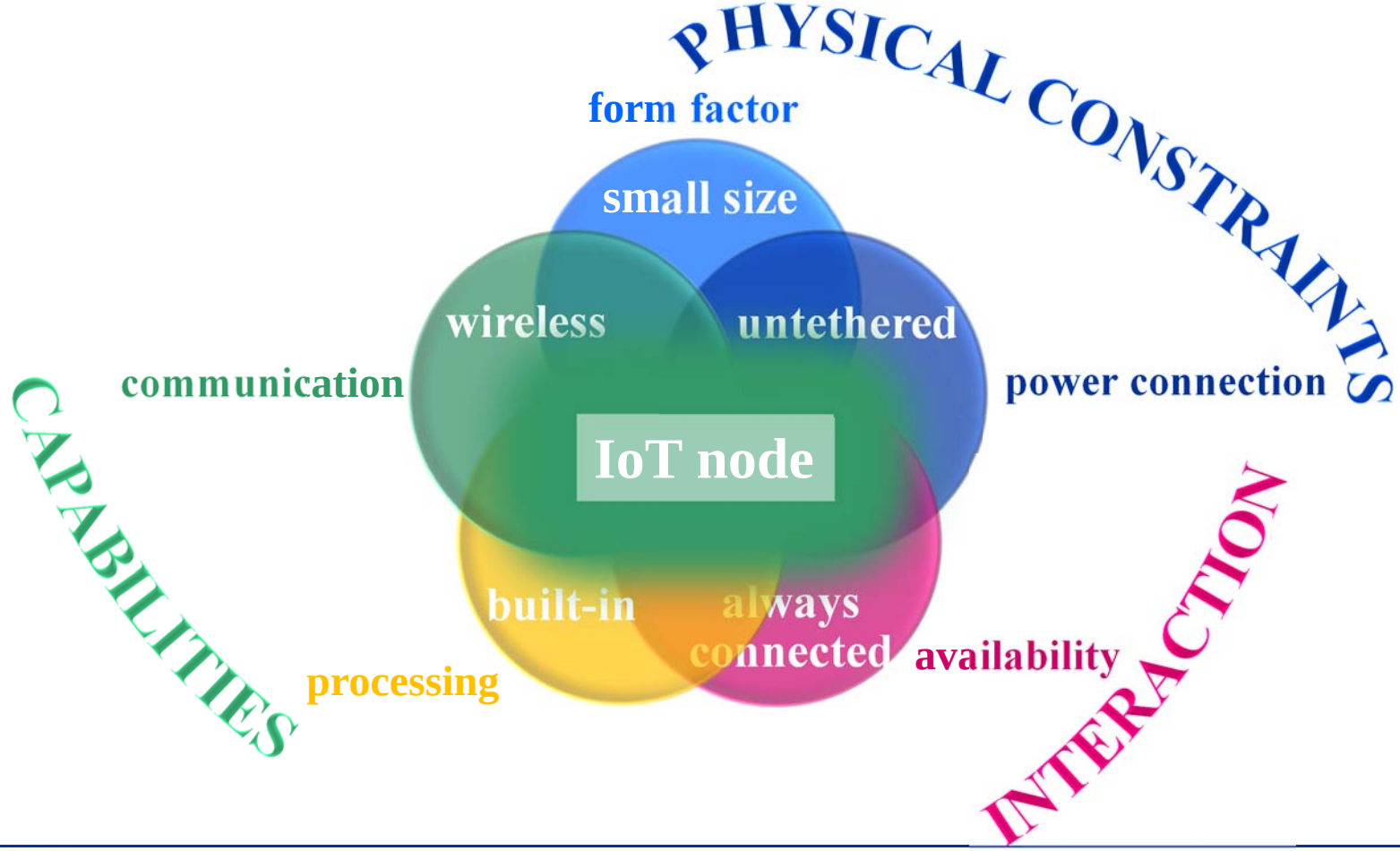
- IoT: the context
- Ultra-low voltage operation
- Design Issues and Solutions at Ultra-Low Voltages: Performance
  - performance
  - leakage
  - variations and resiliency
- Conclusions

Internet of Things: The Context

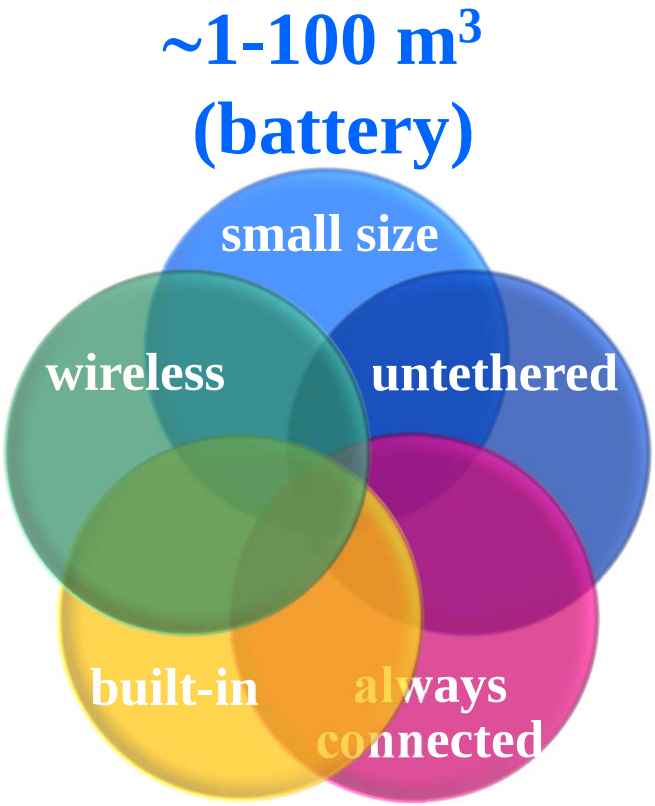


Nodes for the Internet of Things: Peculiarities

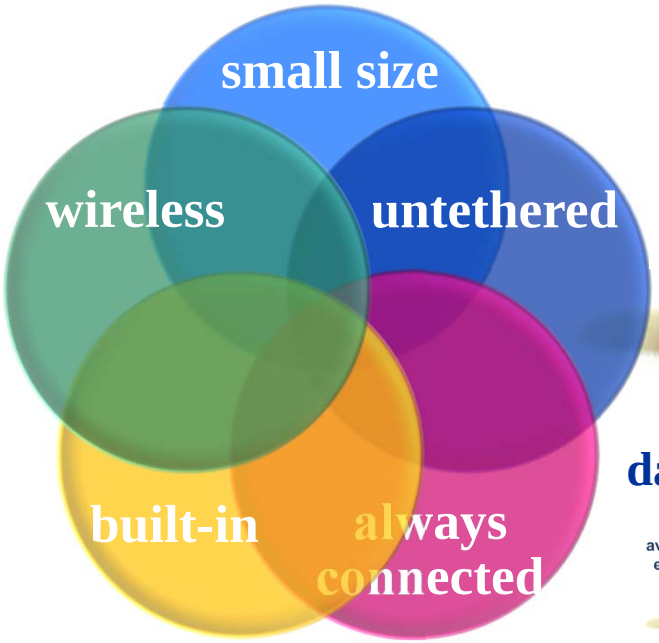
- Specific features of nodes for IoT



- Specific features of nodes for IoT



- Specific features of nodes for IoT

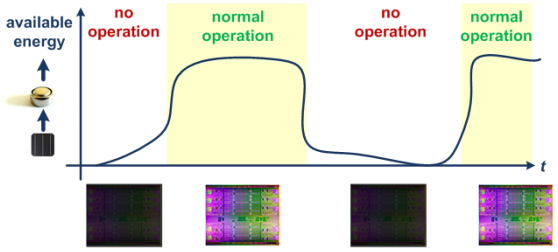


self-powered (limited power)

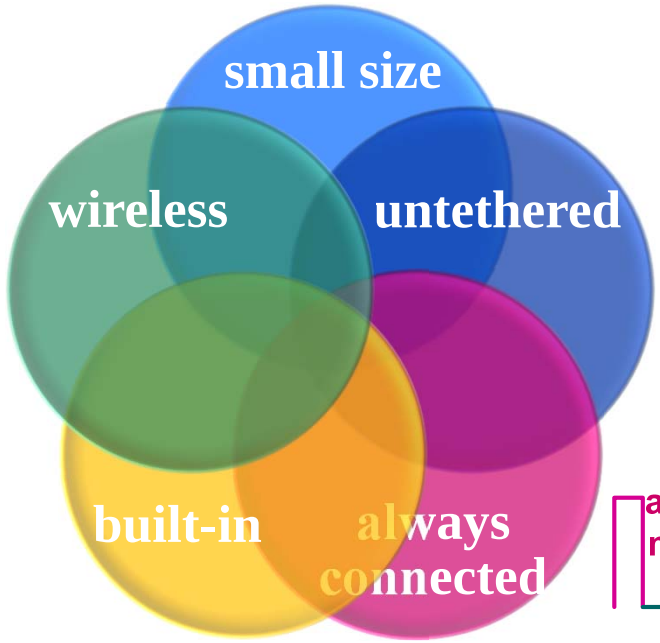
μWs (perpetual)

100 nW (10 yrs)
40 μW (1 week)

dark silicon in time dimension



- Specific features of nodes for IoT



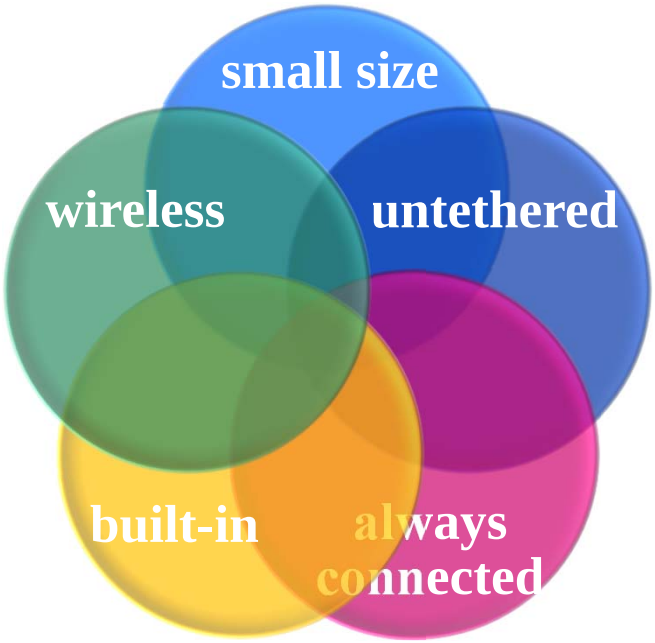
duty cycled

- Specific features of nodes for IoT

computation
repetitive
- leverage specialized HW

- real time: scalable performance is needed

- data logging: less performance, but memory cost



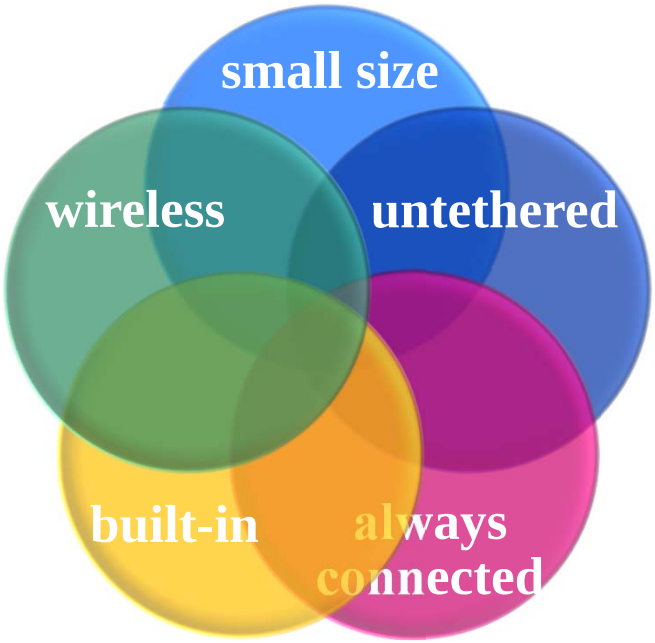
♦ Specific features of nodes for IoT

communication

computation vs communication tradeoff

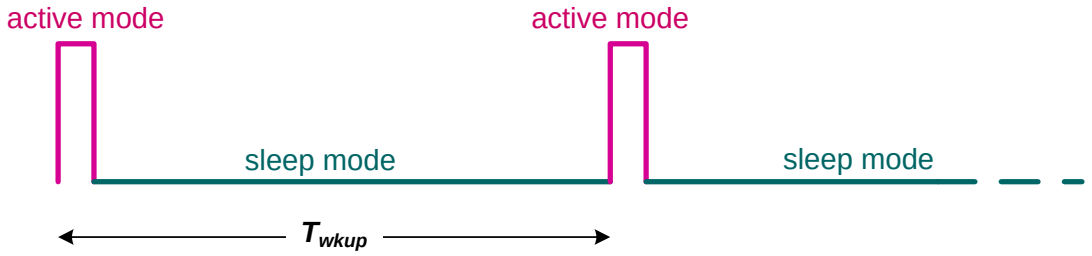
- data representation (compressive sensing, compression)

- limit TX to critical events or significant changes (critical event monitoring)



Power vs Energy

- Duty cycled systems with limited power
  - active only periodically (or on demand) for a short time



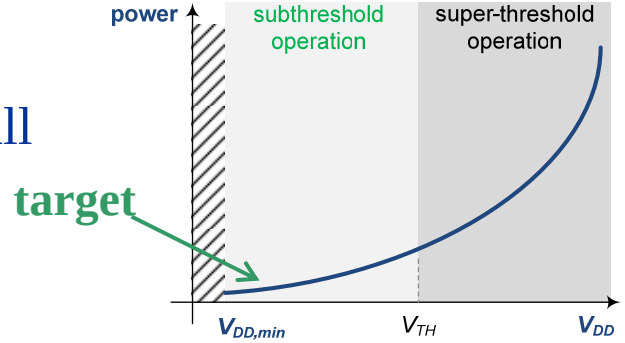
  - partition into **always-on** block (timers, retentive memory) and **duty cycled** blocks (all others, active 0.1-1% of the time)

$$P_{avg} = P_{always-on} + \left(P_{sleep} + E_{active} / T_{wkup}\right)$$

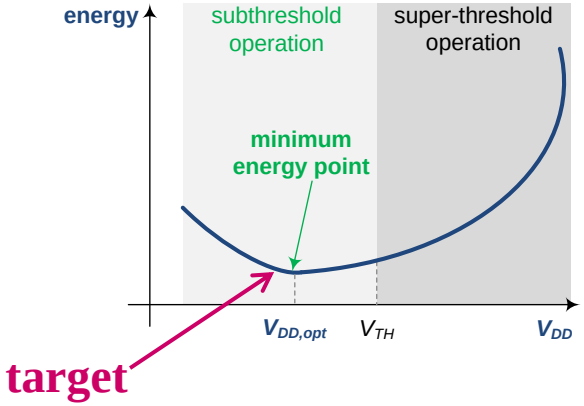
  - $P_{sleep} \neq 0$ if duty cycled block is power gated
  - $P_{sleep} = 0$ if on-chip regulator is shut down

[RJA12] M. Alioto, et Al., "Active RFID: A Perpetual Wireless Communications Platform for Sensors," ESSCIRC 2012

- **minimum power** is the goal for **always-on** block
  - minimize power, essentially leakage (dynamic power very small – little active and slow)



- **minimum energy** per operation is the goal for **duty-cycled** blocks
  - minimize energy per operation (dynamic + leakage energy)
  - 1/*X* duty cycling increases energy budget by *X*



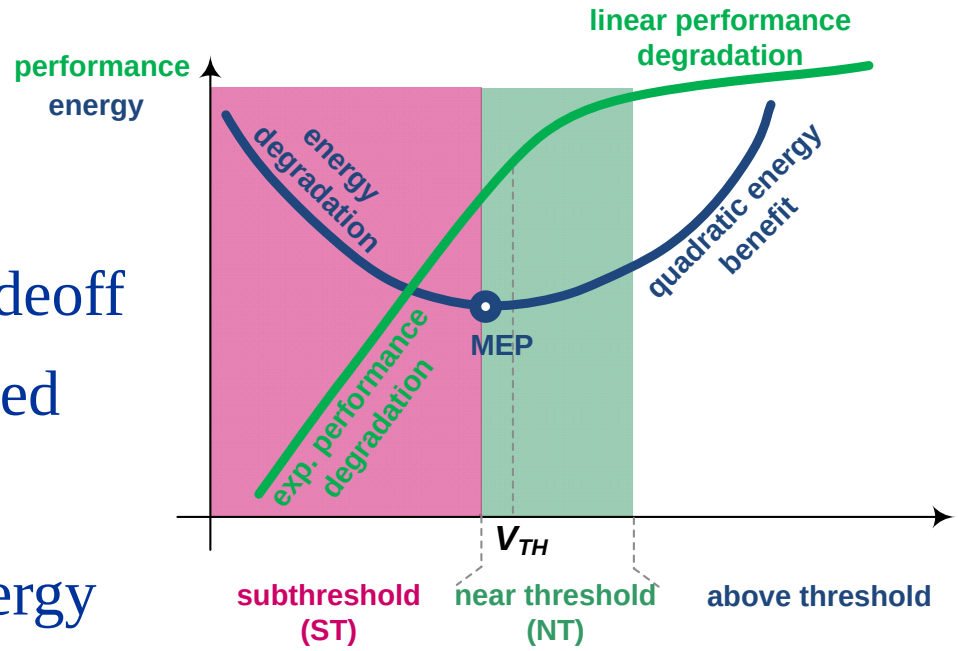
in both cases, ultra-low voltage operation is absolutely needed

Ultra-Low Voltage Operation

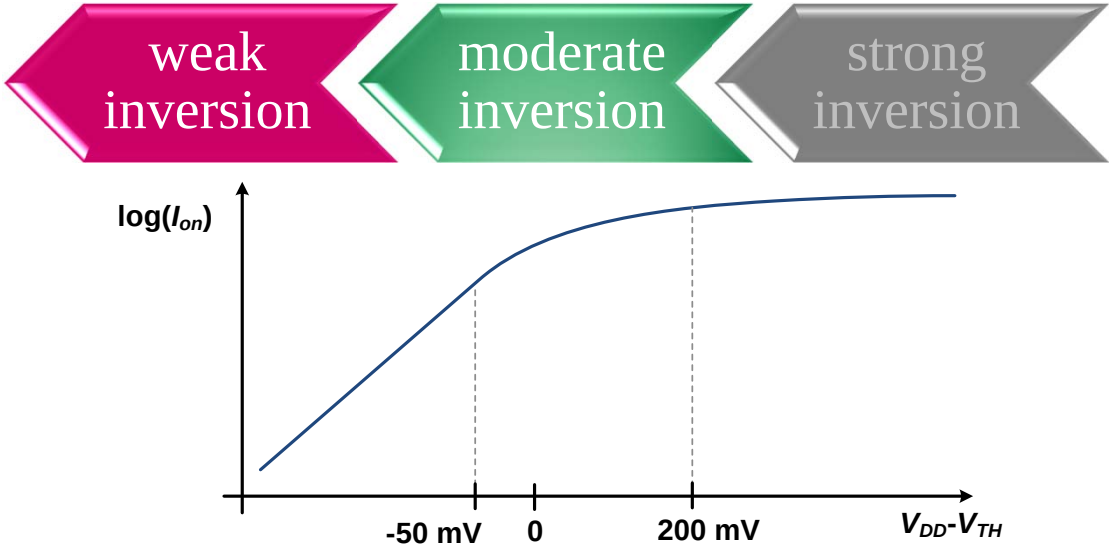
Operation at Ultra-Low Voltages (ULV)

- Voltage scaling is a powerful knob to improve energy efficiency
  - quadratic benefit, if dynamic energy CV^2 dominates
  - performance degradation

- How aggressively should we scale V_{DD}?
  - energy-performance tradeoff
  - **NT**: relatively good speed nearly min. energy,
  - **ST**: low speed, min. energy

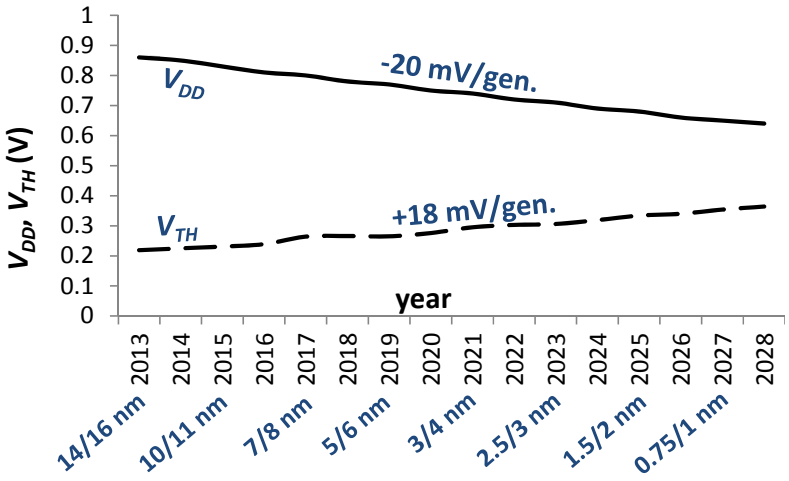


- Transistor operation for IoT systems
  - V_{DD} **close to** or lower than the transistor threshold voltage
  - operation in **moderate** or **weak** inversion
    - region defined by gate overdrive V_{DD}-V_{TH}
    - **NT**: $V_{DD} \in [V_{TH}\text{-}50\text{mV}, V_{TH}\text{+}200\text{mV}] \Rightarrow V_{DD}$**~400-600 mV**
    - **ST**: V_{DD}**<400 mV**

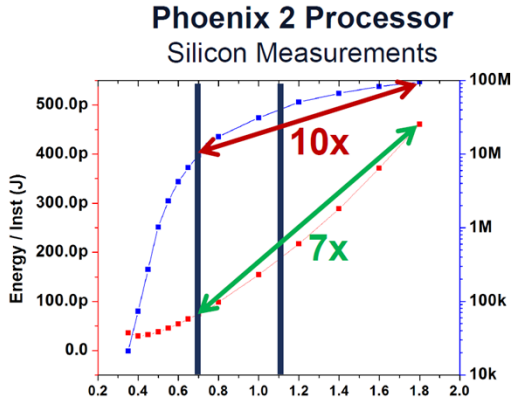


- ITRS: V_{DD}, V_{TH} ~ constant
  - voltage range

    will not change much



- MEP ~ V_{TH}
  - energy ~ minimum (flat minimum)

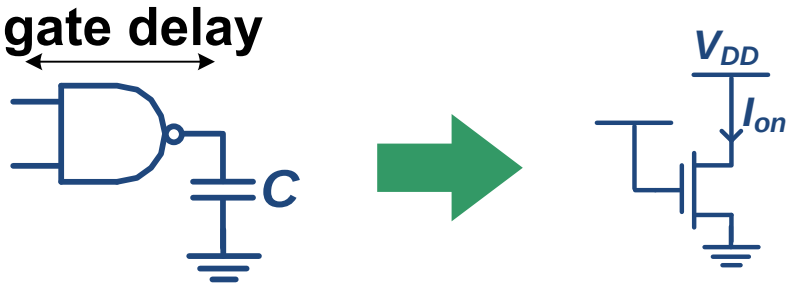


[CGH11] G. Chen, et Al., "A 1 Cubic Millimeter Energy-Autonomous Wireless Intraocular Pressure Monitor," ISSCC 2011

- choice of region
  - **NT**: nearly minimum energy, can have high performance via parallelism (power limited, not area limited)
  - **ST**: minimum power, very lose performance

Performance @ ULV: Transistor On-Current

- Gate delay: $C \cdot V_{DD}/2I_{on}$
  - I_{on} defines performance



- I_{ON} EKV model (transregional) [EV06]
  - Inversion Coefficient vs gate overdrive

$$I_0 = 2 \cdot n \cdot \mu \cdot C_{OX} \frac{W}{L} v_t^2$$

$$IC = \frac{I_{on}}{I_0} = [\ln(e^v + 1)]^2$$

$$v = \frac{V_{DD} - V_{TH}}{2 \cdot n \cdot v_t}$$

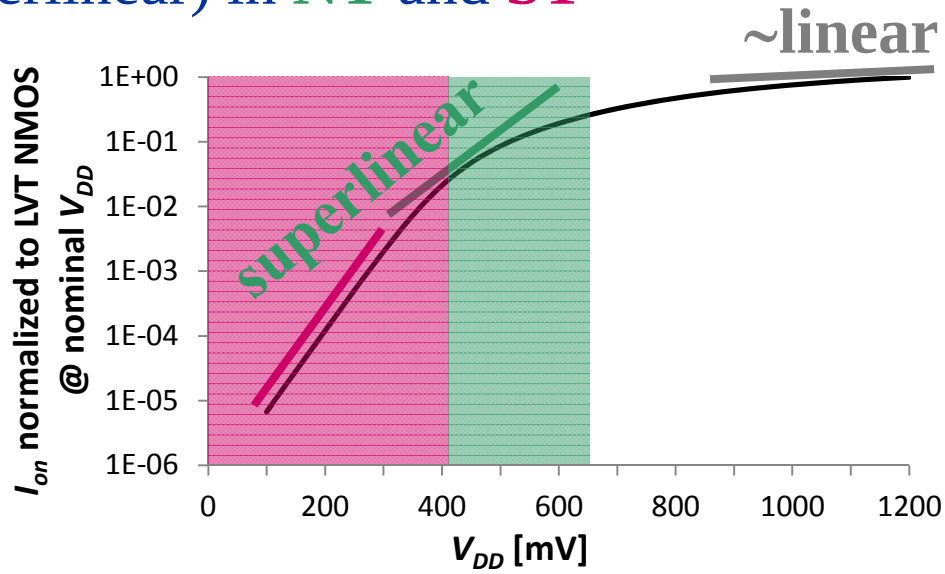
IC < 0.1 **0.1 ≤ IC ≤ 10** **IC > 10**

weak inversion | moderate inversion | strong inversion

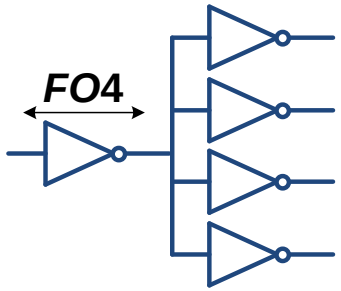
[EV06] C. Enz, E. Vittoz, Charge-Based MOS Transistor Modeling (...), Wiley, 2006

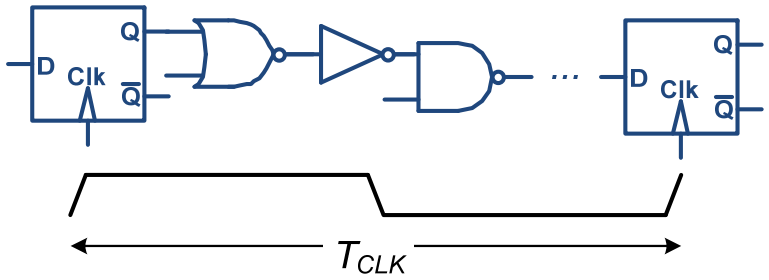
region	V_{DD}	on current
weak inversion	**sub threshold**	$I_{on} \approx I_0 \cdot e^{\frac{V_{DD}-V_{TH}}{n \cdot v_t}}$
moderate inversion	**near threshold**	$I_{on} \approx 0.54 \cdot I_0 \cdot (v + 0.88)^2$
strong inversion	above threshold	$I_{on} \approx I_0 \cdot [v]^{\alpha}$ $(\alpha \sim 1)$

- I_{ON} vs V_{DD}: steep (superlinear) in **NT** and **ST**
  - performance sensitive to both V_{DD} and V_{TH}
  - choice of V_{TH} really critical
  - sensitivity to V_{DD}, V_{TH} variations...



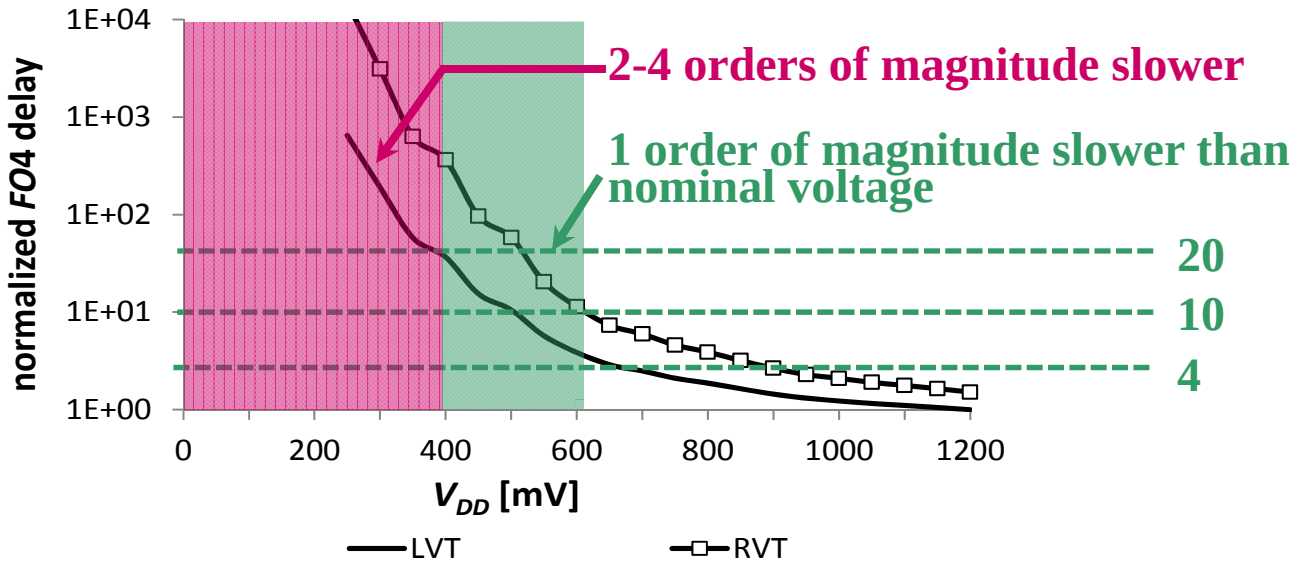
- Performance: fan-out-4 delay (*FO4*)
  - speed of technology and (μ)architecture





effective logic depth

$$T_{CK} = LD_{eff} \cdot FO4$$



- Leveraging high sensitivity of I_{on} to V_{DD} @ NT
  - V_{DD} powerful knob to dynamically improve performance
    - performance improvement due to ΔV=100-mV boosting

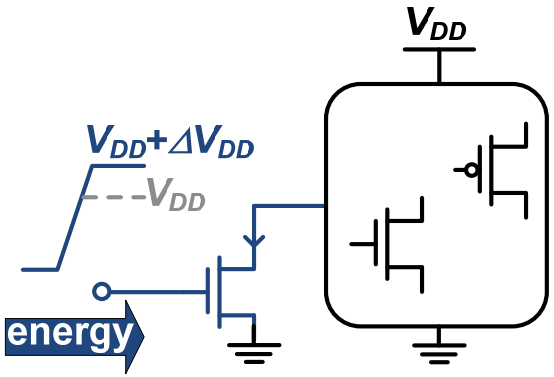
V_{DD}	*FO4* improvement
200 mV	**7.5X**
400 mV	**3.6X**
600 mV	**1.6X**
800 mV	1.2X
1 V	1.1X

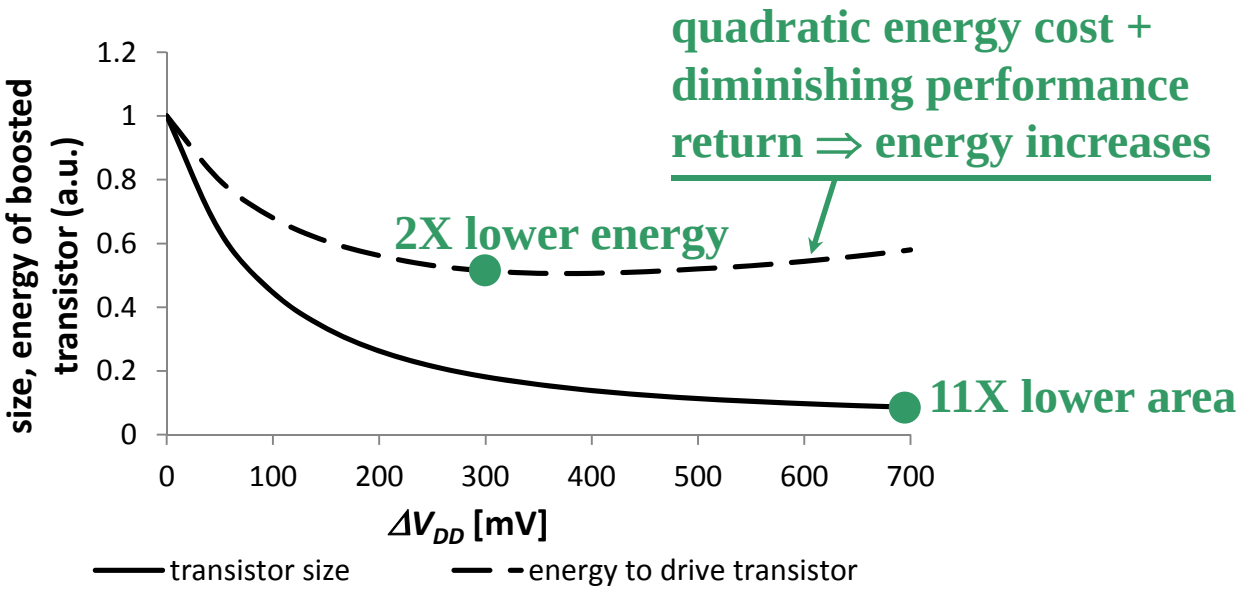
ST: 8X improvement/100 mV

NT: 2-4X improvement/100 mV

    - **ST**: more effective, pretty useless (low performance anyway)
    - **NT**: effective and low energy cost (ΔV is small fraction of V_{DD})
    - **ST** and **NT**: can effectively compensate performance variations

- Selective voltage boosting largely improves area and energy efficiency at **NT** and **ST**
  - sizing has a linear impact on transistor strength
  - boosting: superlinear ⇒ smaller transistor at iso-strength
  - example at **NT** (V_{DD}=500 mV)



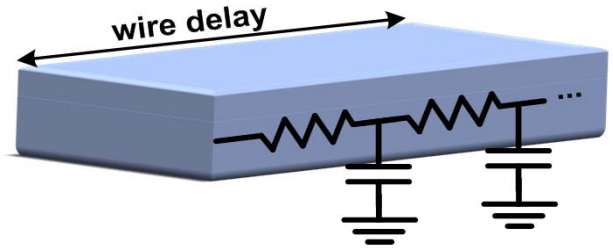


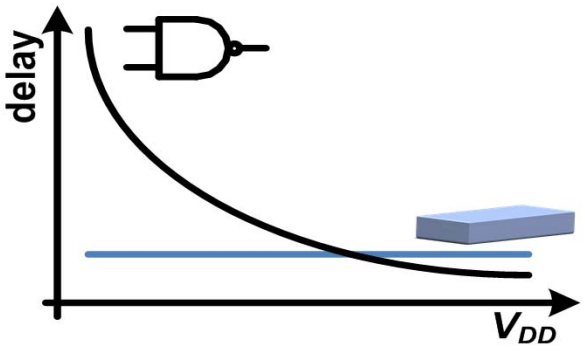
Design Issues and Solutions at Ultra-Low Voltages: Performance

Performance Degradation

- Performance at **NT**/**ST** worse than nominal voltage
  - can be acceptable in practical situations
    - tasks of IoT nodes are often relatively simple
    - example of typical throughput: few **hundreds of Mops/s** (e.g., video processing) down to **kops/s** (e.g., temperature monitoring)

- Wire delay at **NT**/**ST**
  - gate delay increases at ULV (>10X)
  - wire delay is constant
  - gate delay much less critical than nominal V_{DD}
    - design as in the "good old days"





- If higher performance is sometimes needed...
  - wide voltage scaling
    (e.g., from 400 mV to 1.2 V)

  - large performance improvement (10X)
  - energy/op increases quadratically (9X)
  - tolerable if occasional

$$E_{dyn} \propto C \cdot V_{DD}^2$$

  - optimized for NT

⇒ **energy/performance tradeoff at 1.2 V is degraded**

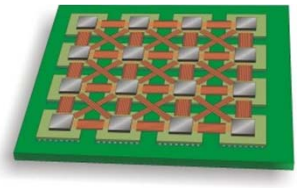
Design Characteristics	Synthesis Corner	
	0.5V, 80MHz	1.05V, 650MHz
Performance Range (0.5V-1.05V)	1X-8X	0.6X-6.5X
Leakage Power	1X	0.8X
Total Device Width	1X	0.78X

Design Corner Evaluations

[J12] S. Jain, et Al., IEEE ISSCC Dig. Tech. Papers, pp. 66–67, Feb. 2012

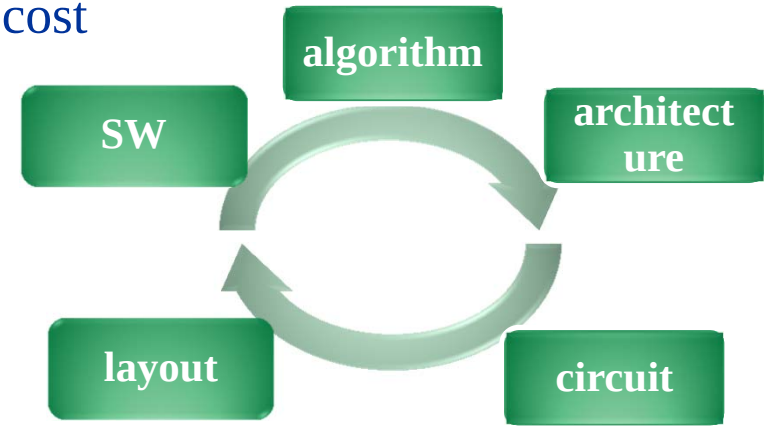
area

performance, energy efficiency

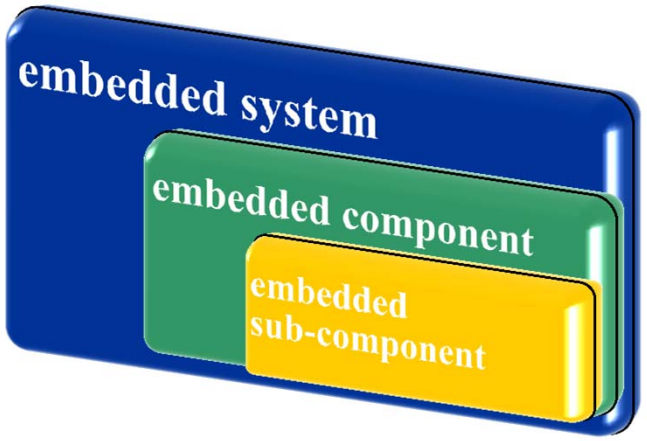


- trading off area for performance: parallelism
  - **intra-chip communication** limits perf./energy gains
  - **specificity of task** can be leveraged for better balance of computation/communication cost
  - **across-level design** is required to manage these tradeoffs

NT



- trading off area for performance: specialized HW
  - example: FFT, Java processor, AES, MPEG decoder...
  - specialized HW has better performance/energy tradeoff than general-purpose



  - larger benefit for **recurrent and specific tasks**

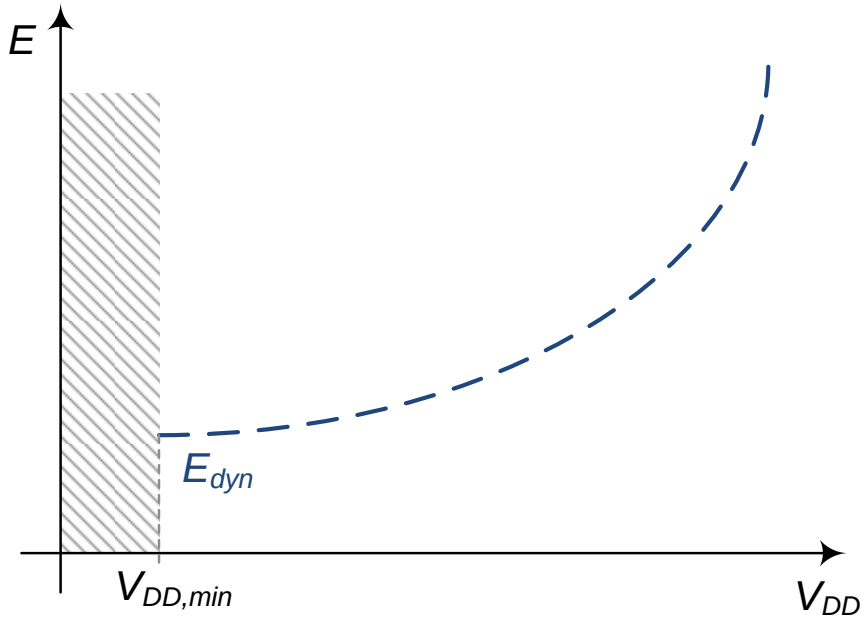
Design Issues and Solutions at Ultra-Low Voltages: Leakage

Energy vs V_{DD}

- If dynamic energy per operation dominates:

$$E_{dyn} = \alpha_{SW} \cdot C \cdot V_{DD}^2$$

  - reduce V_{DD} as much as possible
    - energy reduction limited by $V_{DD,min}$ (defined by robustness issues, very different for logic and memory)



V_{DD} (mV)	V_{DD}^2 energy saving	including $C_g(V_{DD})$
200 mV	36X	54X
400 mV	**9X**	**11.6X**
600 mV	**4X**	**4.4X**
800 mV	2.2X	2.4X
1 V	1.4X	1.4X
1.2 V	1X	1X

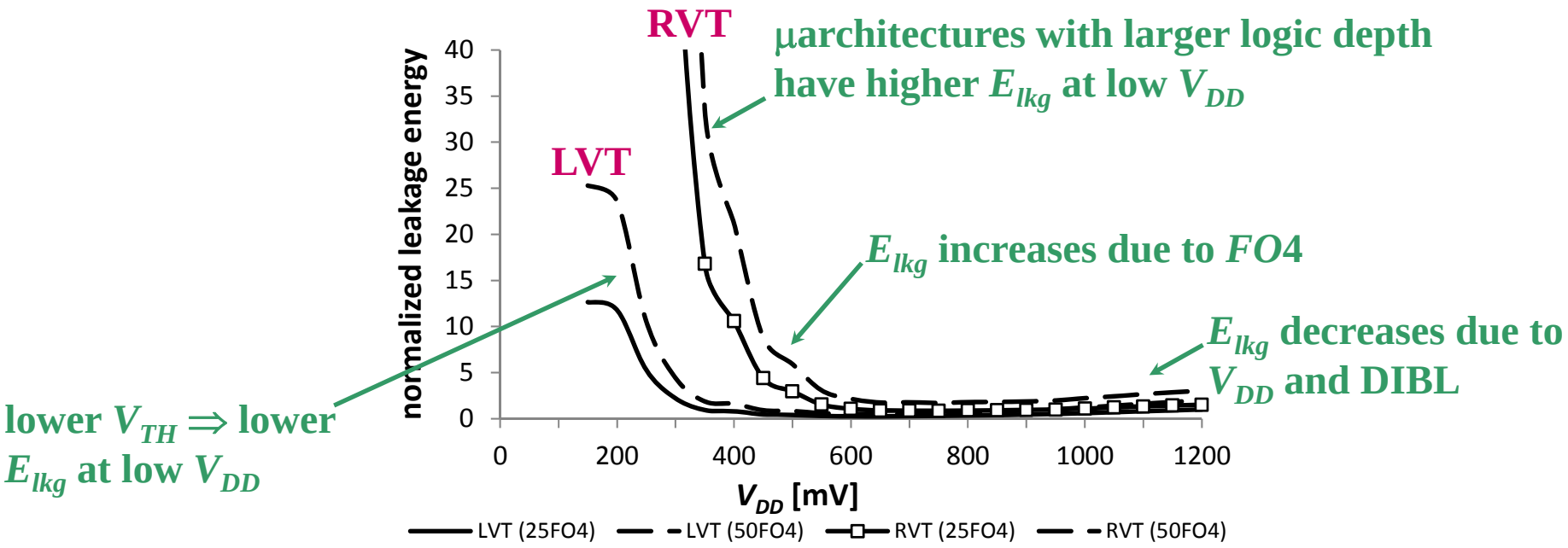
- If leakage energy per operation dominates:

$$E_{lkg} = V_{DD} \cdot I_{off} \cdot FO4 \cdot LD_{eff} \cdot CPO$$

architecture
ckt
technology

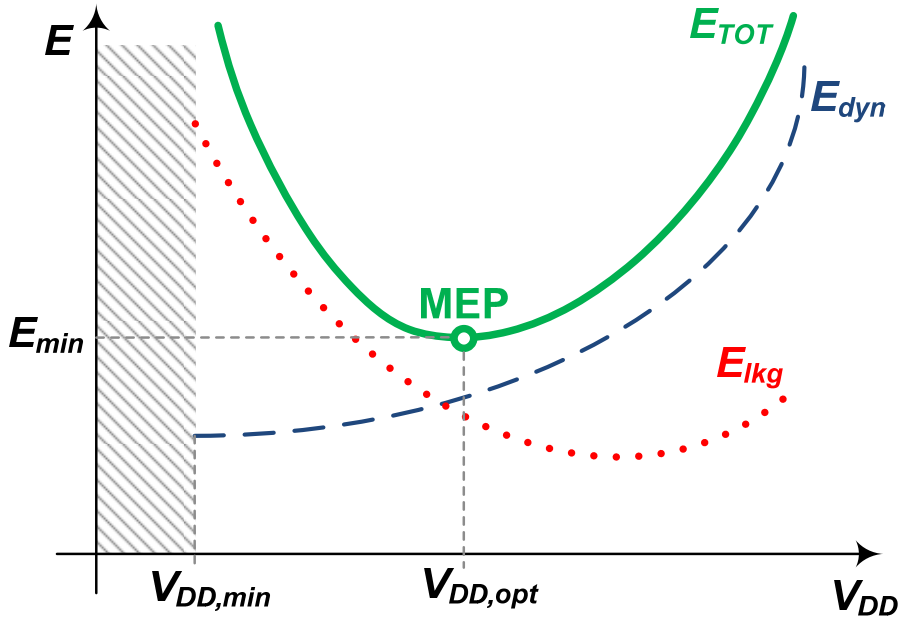
cycles/op

  - at **NT** and **ST**, increases significantly ($FO4$ increase)
    - example in 28nm:



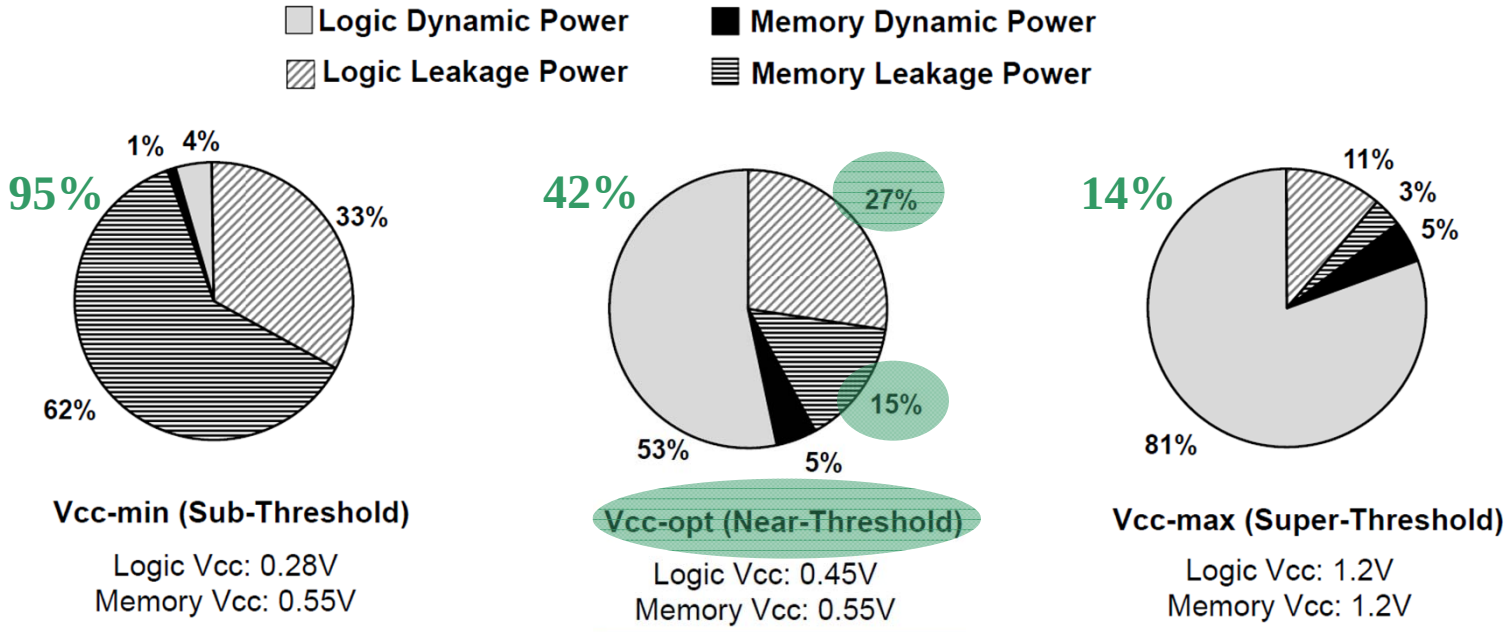
Minimum Energy Point

- Combining dynamic and leakage energy
  - minimum energy point (**MEP**)
    - relatively **flat** (V_{DD} mainly set by performance target)
    - can lie in either **NT** or **ST**
    - **time varying**: depends on temperature, data set...



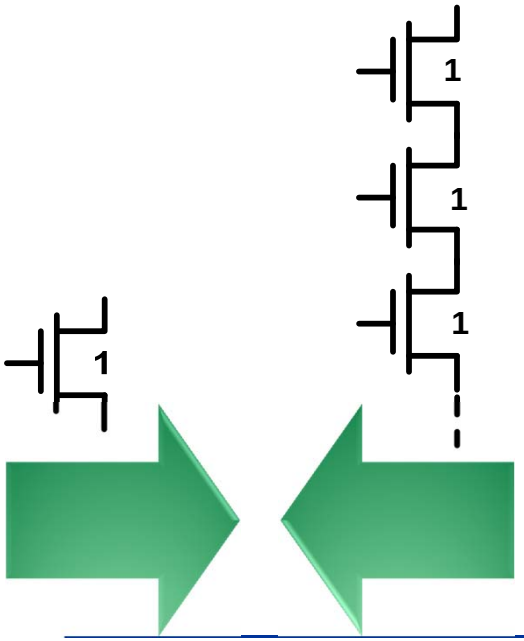
- Leakage energy takes up increasingly larger fraction of total energy at lower V_{DD}
  - at low V_{DD}, leakage energy increases exponentially, dynamic energy decreases quadratically
  - ex.: processor with L1 cache ($V_{DDcache,min}$=0.55 V)

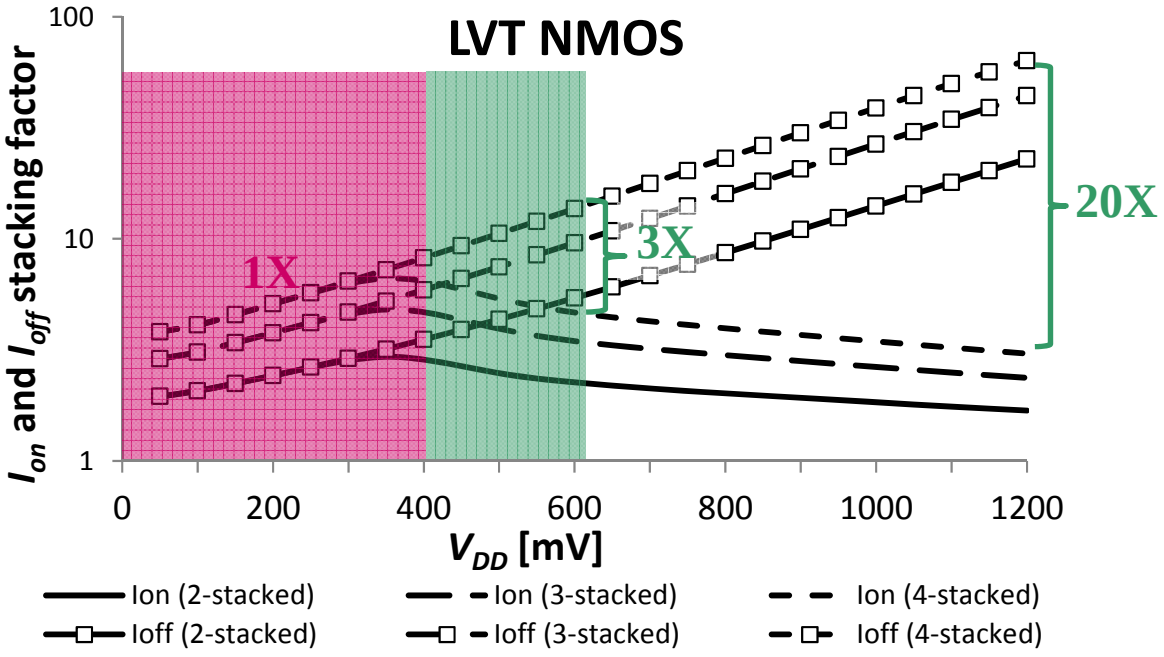
[J12] S. Jain, et Al., IEEE ISSCC Dig. Tech. Papers, pp. 66–67, Feb. 2012



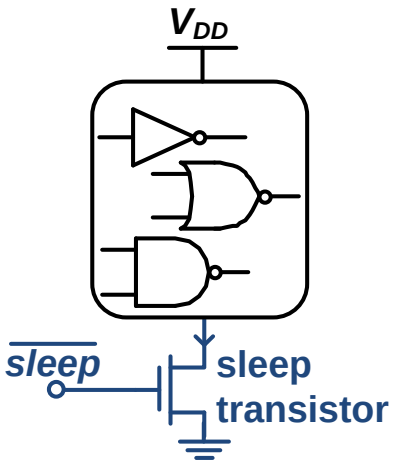
Traditional Techniques for Low Leakage

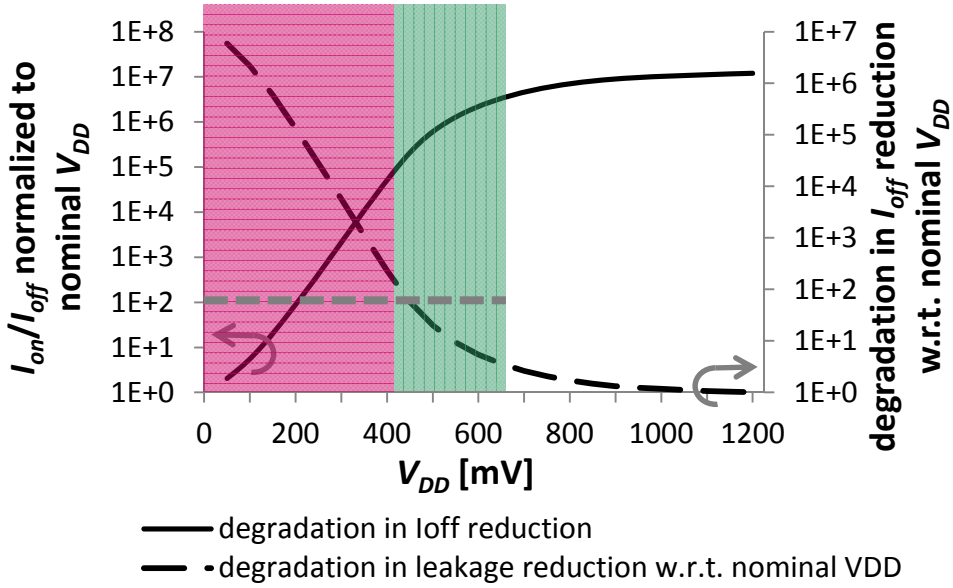
- Leakage is truly critical (process not enough)
  - large, limits energy reduction
- Several traditional circuit techniques do not work...
  - **transistor stacking** is ineffective





- **power gating** is much less effective (I_{on}/I_{off} degradation)





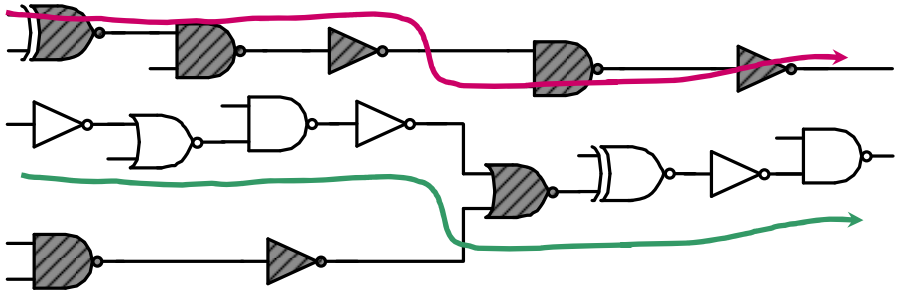
- typical leakage reduction: 10-100X
- **NT**: small leakage reduction, **ST**: no leakage reduction at all
- solution: boost gate voltage of sleep transistor (increases I_{on}/I_{off})

selective voltage boosting

- **multi-V_{TH} actually degrades energy efficiency**
  - delay sensitive to $V_{TH} \Rightarrow$ critical path changes at scaled V_{DD}

V_{DD}=0.6 V

V_{DD}=0.4 V



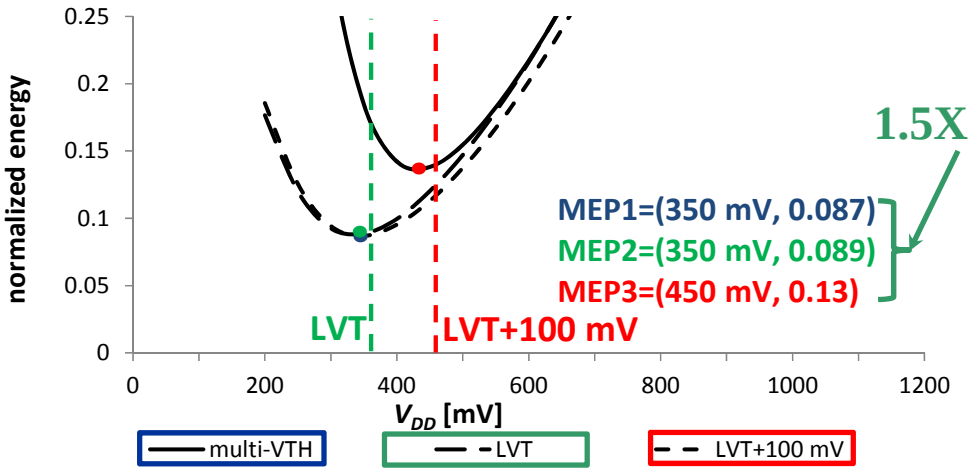
HVT

LVT

$$E_{lkg} = V_{DD} \cdot I_{off} \cdot T_{CK}$$

larger than single-(high)V_{TH}

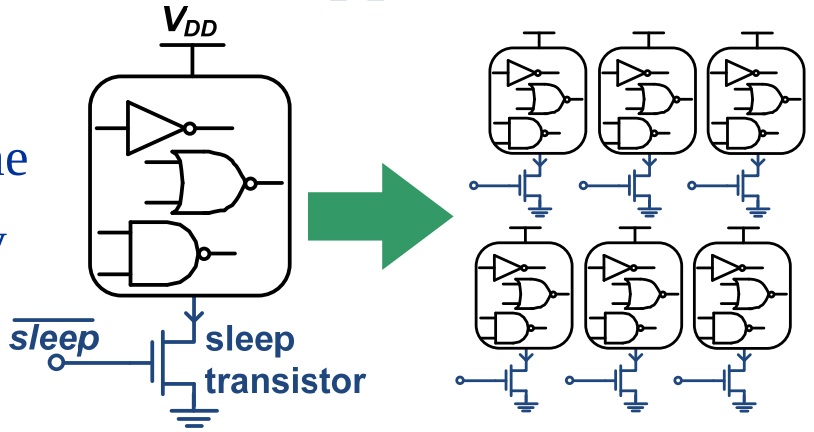
larger than single-(low)V_{TH}



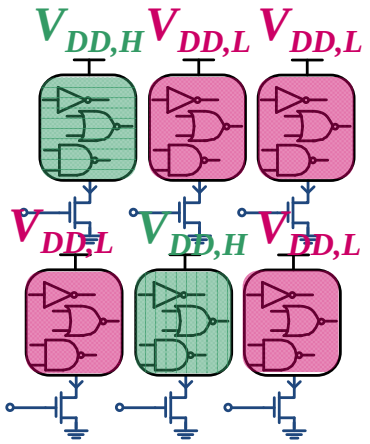
  - double energy penalty compared to single-V_{TH}
  - double energy penalty

Counteracting Leakage at ULV

- Leakage reduction at ULV: alternative approaches
  - fine-grain power gating
    - disable unused blocks at runtime
    - small ⇒ quickly and frequently
    - **overhead**: multiple sleep transistors, isolation, control
    - lower control overhead via ckt/architectural support for SW



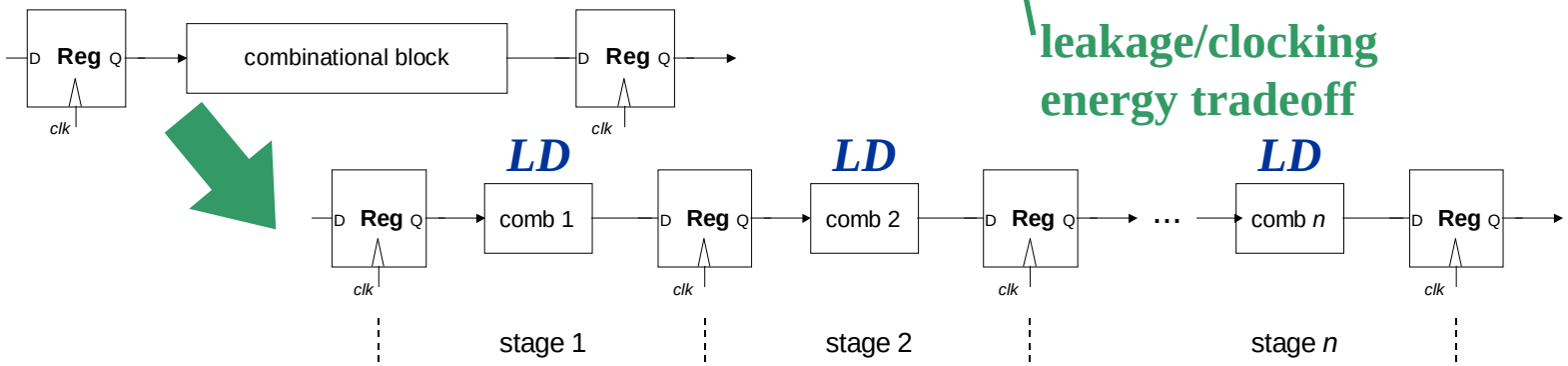
  - fine-grain voltage domains
    - E_{lkg} reduced at lower V_{DD} (e.g., 2X/100 mV)
    - selectively reduce V_{DD} wherever possible (slower)
    - similar considerations as power gating



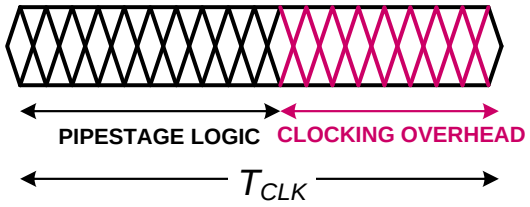
- microarchitecture-circuit co-design: pipelining

$$E_{lkg} = V_{DD} \cdot I_{off} \cdot FO4 \cdot \boldsymbol{LD_{eff}} \cdot CPO$$

leakage/clocking energy tradeoff

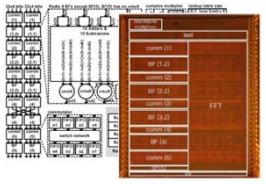


- use **deep pipelines** + refined circuit techniques/methodologies to deal with clocking overhead



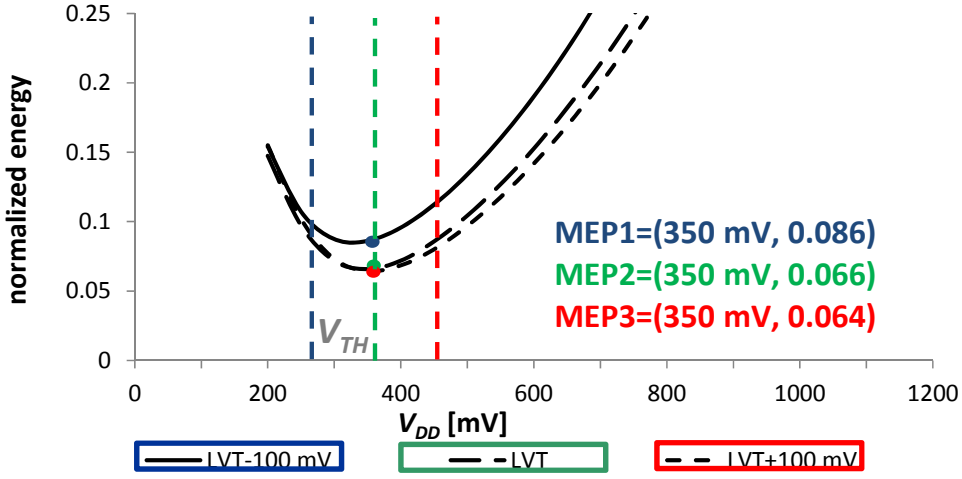
- example: 17FO4/stage in FFT engine

(30MHz @ 0.27V, 4X less energy than state of the art)



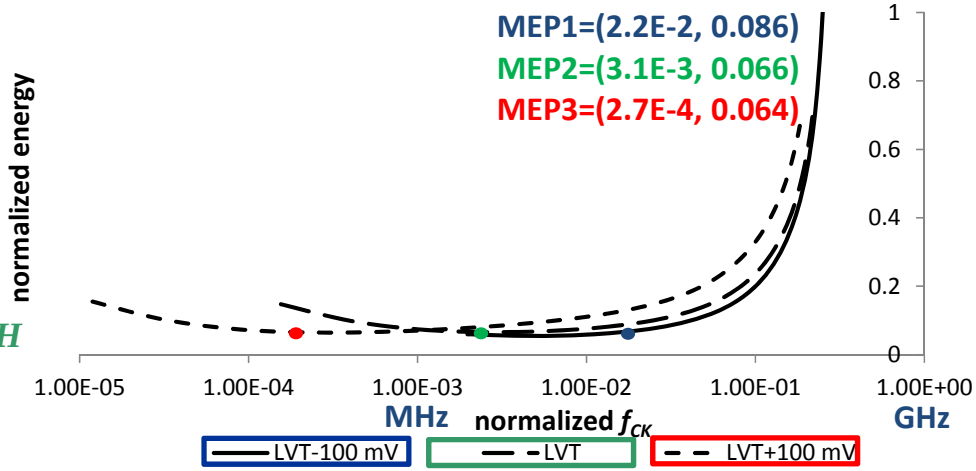
[AJC11] M. Seok, et Al. "A 0.27V, 30MHz, 17.7nJ/transform 1024-pt complex FFT core with super-pipelining," ISSCC 2011

- V_{TH} selection
  - energy-optimal $V_{DD,opt}$ **is independent of V_{TH}**
  - choose V_{TH} according to performance target



[A12] M. Alioto, "Ultra-Low Power VLSI Circuit Design Demystified and Explained: A Tutorial," IEEE TCAS-I, Jan. 2012.

  - performance: changes by 10X per V_{TH} flavor
  - **strong process-μarch. interaction through V_{TH}**

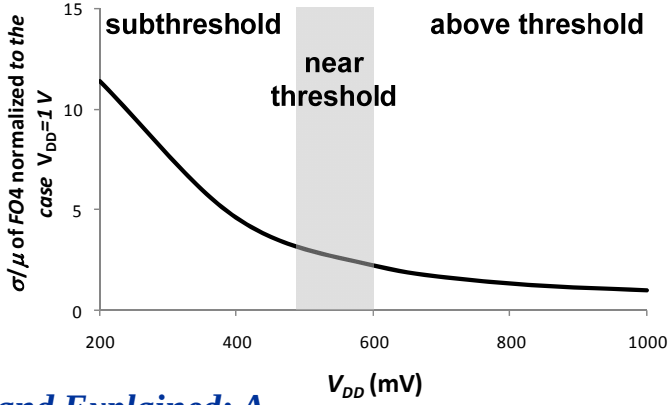


- body biasing
  - makes sense in FDSOI (V_{TH} sensitivity to V_{BB} large enough)

Design Issues and Solutions at Ultra-Low Voltages: Variability and Resiliency

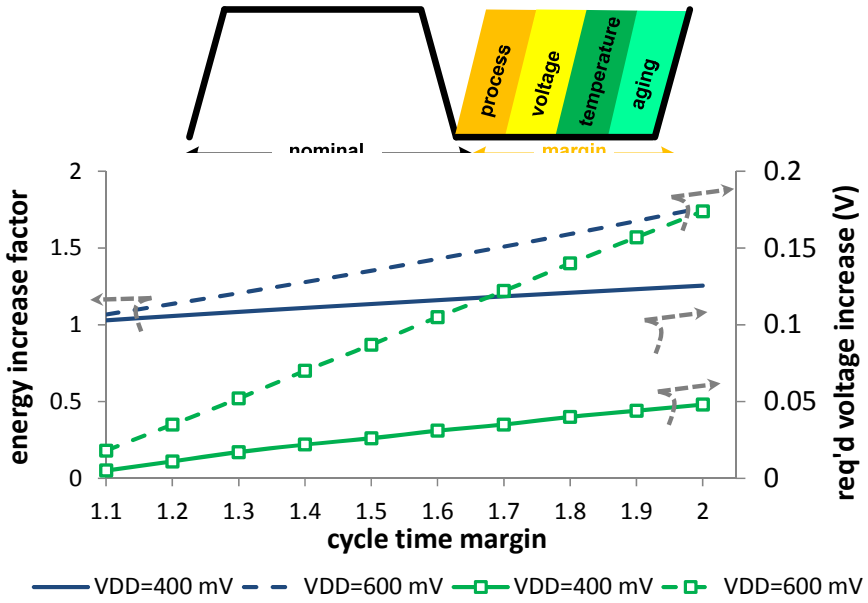
Variations: Why do They Matter?

- Resiliency degraded at ULV
  - process/voltage/temperature
    - larger than at nominal V_{DD}
    - ageing, soft errors...



[A12] M. Alioto, "Ultra-Low Power VLSI Circuit Design Demystified and Explained: A Tutorial," IEEE TCAS-I, Jan. 2012.

  - design margining
    - cycle margin
      (20-30% @ full V_{DD})
    - **degrades performance AND energy efficiency**
    - large energy penalty



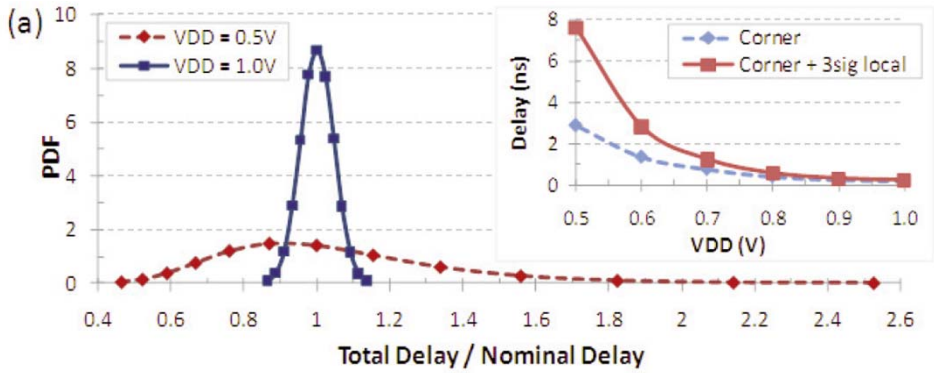
Process Variations

- Variability of delay (mainly due to I_{on})
  - random variations are dominant (area-mismatch tradeoff)
  - two negative effects arise at NT [GIS11]

Random Dopant Fluctuation, Line Edge Roughness...

Larger variability σ/μ

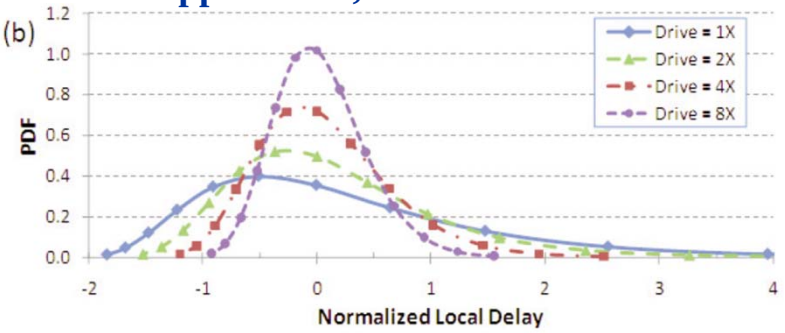
- due to larger sensitivity of I_{on} to V_{TH}



[GIS11] G. Gammie, et Al., "A 28nm 0.6V low-power DSP for mobile applications," *ISSCC* 2011

PDF heavily non-Gaussian

- in subthreshold tends to lognormal
- right skewed ⇒ average > nominal
- larger no. of σs for given yield



♦ Semi-quantitative analysis of I_{on} variations

Larger variability σ/μ

Above threshold

$$I_{on} \propto \left(1 - \frac{\Delta V_{TH}}{V_{DD} - V_{TH}}\right)$$

$$\Rightarrow \frac{\sigma_{I_{on}}}{\mu_{I_{on}}} = \frac{\sigma_{V_{TH}}}{V_{DD} - V_{TH}}$$

ex. V_{DD}=0.9 V, V_{TH}=0.36 V, σ_{VTH}=35 mV

⇒ 6.5%

Subthreshold

$$I_{on} \propto e^{-\frac{\Delta V_{TH}}{n \cdot v_t}}$$

$$\Rightarrow \frac{\sigma_{I_{on}}}{\mu_{I_{on}}} = \sqrt{e^{\left(\frac{\sigma_{V_{TH}}}{n \cdot v_t}\right)^2} - 1}$$

ex. σ_{VTH}=35 mV, n=1.35

⇒ 131% (20X)

♦ at NT, σ/μ is intermediate (**~4-5X**)

Larger no. of σ

♦ no. of std deviations (yield)

- ♦ at NT: somewhat intermediate
- ♦ ex. @ 3σ for single gate:

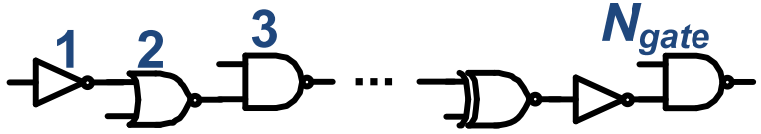
margin = 10σ/μ = 300%

YIELD	Gaussian	lognormal
84%	σ	e·σ
97.7%	2σ	$e^2 \cdot \sigma \approx 7.4\sigma$
99.9%	3σ	$e^3 \cdot \sigma \approx 20.1\sigma$

♦ Variability/leakage tradeoff unavoidable

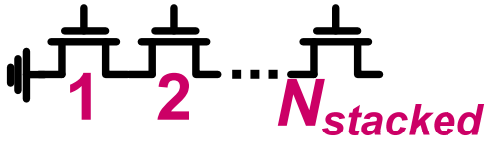
- ♦ averaging effect reduces variability by

cascading ♦ cascaded logic gates



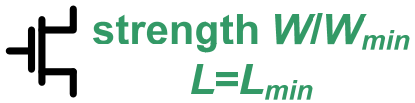
$$\frac{\sigma}{\mu} = \frac{1}{\sqrt{N_{gate}}}$$

stacking ♦ stacked transistors



$$\frac{\sigma}{\mu} = \frac{1}{\sqrt{N_{stacked}}}$$

strength ♦ larger transistors



$$\frac{\sigma}{\mu} = \frac{1}{\sqrt{strength}}$$

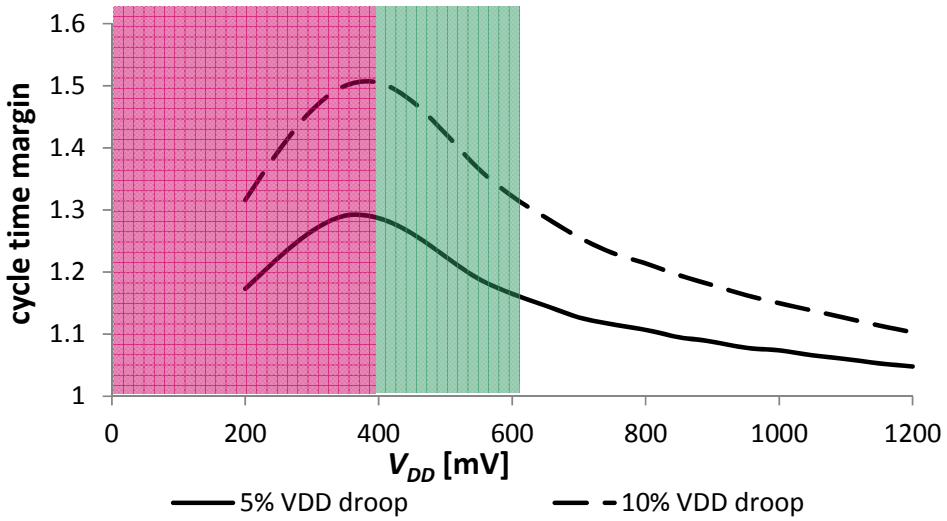
- ♦ path delay variability from *FO4* variability (min. sized)

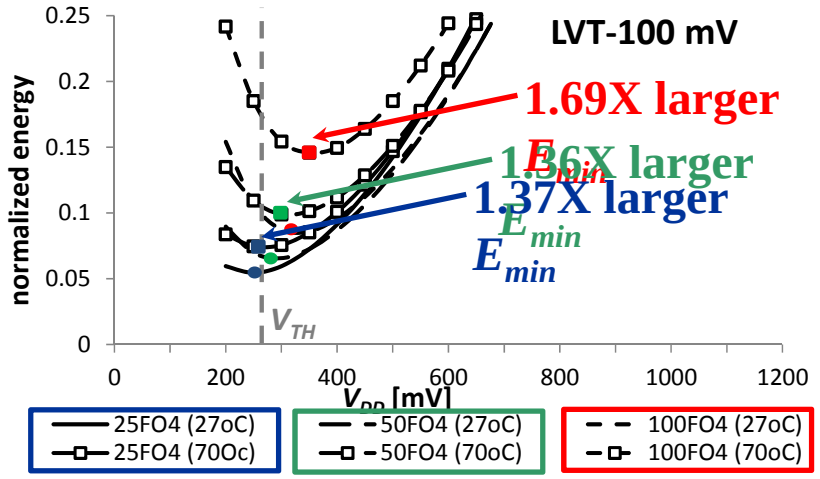
$$n_\sigma \frac{\sigma_{pathdelay}}{\mu_{pathdelay}} = n_\sigma \sqrt{\left(\frac{\sigma_{FO4,D2D}}{\mu_{FO4,D2D}}\right)^2 + \frac{\left(\frac{\sigma_{FO4,random}}{\mu_{FO4,random}}\right)^2}{N_{gate} \cdot strength \cdot N_{stacked}}}$$

[MWA10] M. Merrett, et Al., "Design Metrics for RTL Level Estimation of Delay Variability Due to Intradie (Random) Variations," ISCAS 2010

Voltage/Temperature Variations

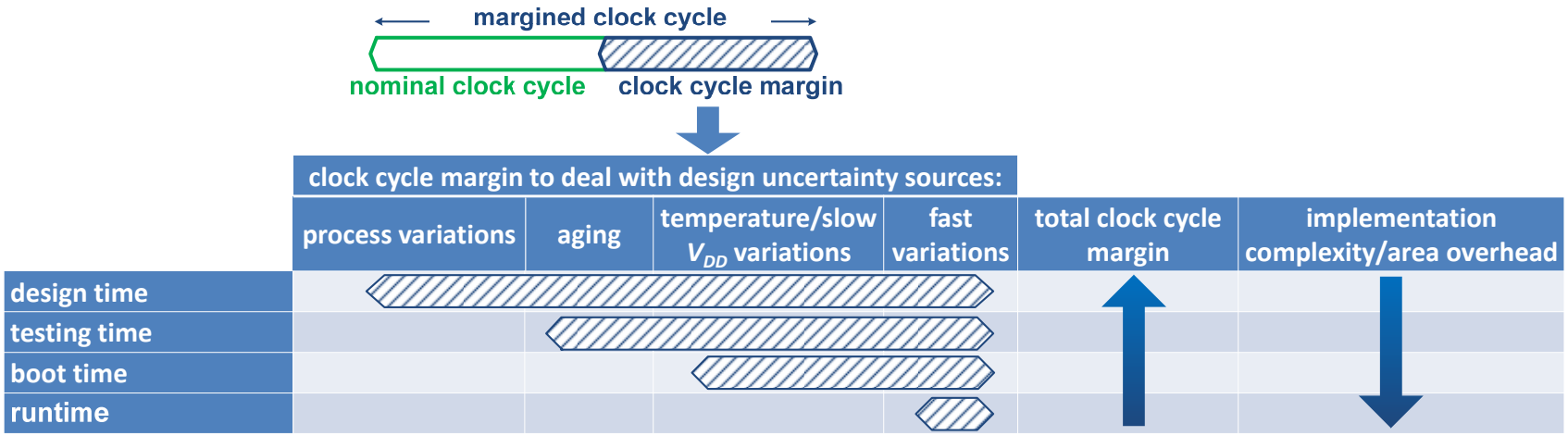
- Delay is very sensitive to V_{DD} and temperature
  - voltage: up to 30-50% margin
  - temperature: up to 2X margin
  - **NT** systems require **adaptive schemes**
    - sense V_{DD} and T and adjust clock cycle
    - can compensate **slow variations** (margin needed for fast)





Clock Cycle Margin Reduction/Elimination

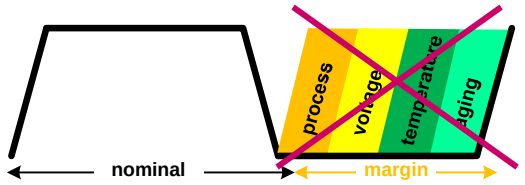
- Compensation of variations at different times



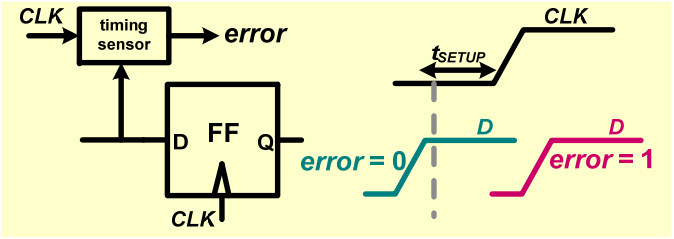
  - tradeoff between energy cost of margining vs energy cost to reduce amount of margin
  - large variations at **NT**/**ST**: runtime compensation usually required
  - detect timing errors/correct: minimum or no margin at all

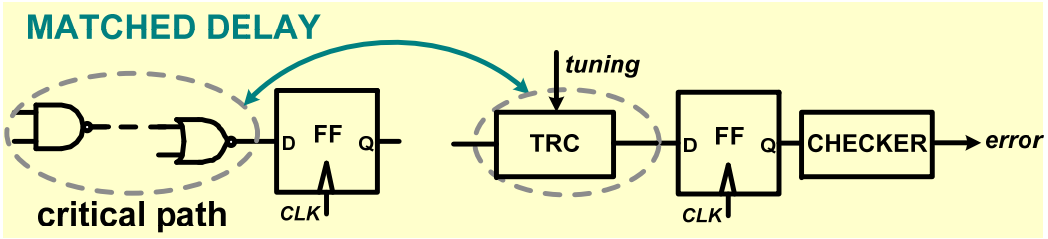
Margin Elimination: Timing Error Detection

- Reduce/eliminate worst-case margin by catching delay faults



  - correct at run-time, tune to compensate actual variations
    - can speculatively reduce energy (if critical path is infrequent)





In-situ monitoring

- P, V, T, aging, fast variations
- no margin
- invasive, limited tuning

Fault prediction (Tunable Replica Circuit)

- partially: P, V, T, aging, fast (not soft errors)
- needs some margin (false positives, mimics only critical path)
- little invasive, tuning required, low overhead

Margin Elimination: Error Correction

- Faults can be corrected at various levels
  - across-level design/optimization/control needed

faster correction

SW → Architecture → Microarchitecture → Circuit

less HW resources

Circuit → Microarchitecture → Architecture → SW

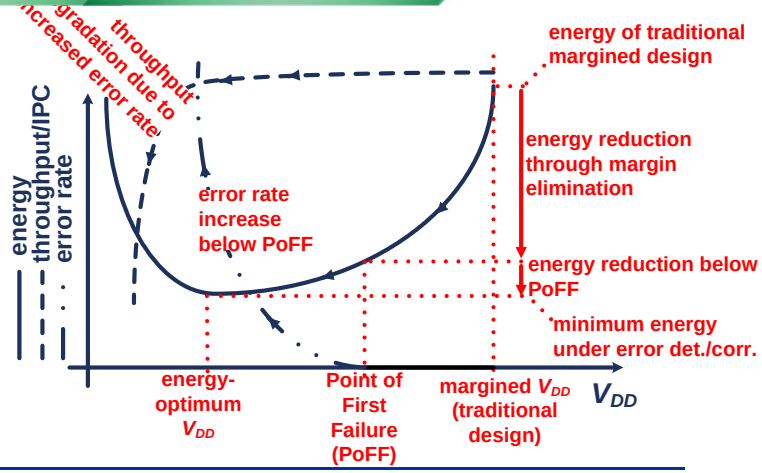
lower energy/performance penalty

SW → Architecture → Microarchitecture → Circuit

- energy overhead

$$E = E_{correction} \cdot error\ rate + f \cdot C \cdot V_{DD}^2$$

- **testing is painful** (long, tuning)



A Step Forward: Sub-Cycle Detection/Correction

- Most existing approaches are cycle-based

	Area	Power	Complexity	Recovery Time [a]
Clock Gating	High	High	Impractical	1 cycle
Counterflow	Medium	Medium	Medium	1 + (N – 1) cycles
Micro-Rollback	High	High	High	N cycles
Multiple Issue	Low	Low	Low	2N + 2 cycles

[CKM11] J. Crop, et Al., "Error Detection and Recovery Techniques for Variation-Aware CMOS Computing: A Comprehensive Review" JLPEA 2011

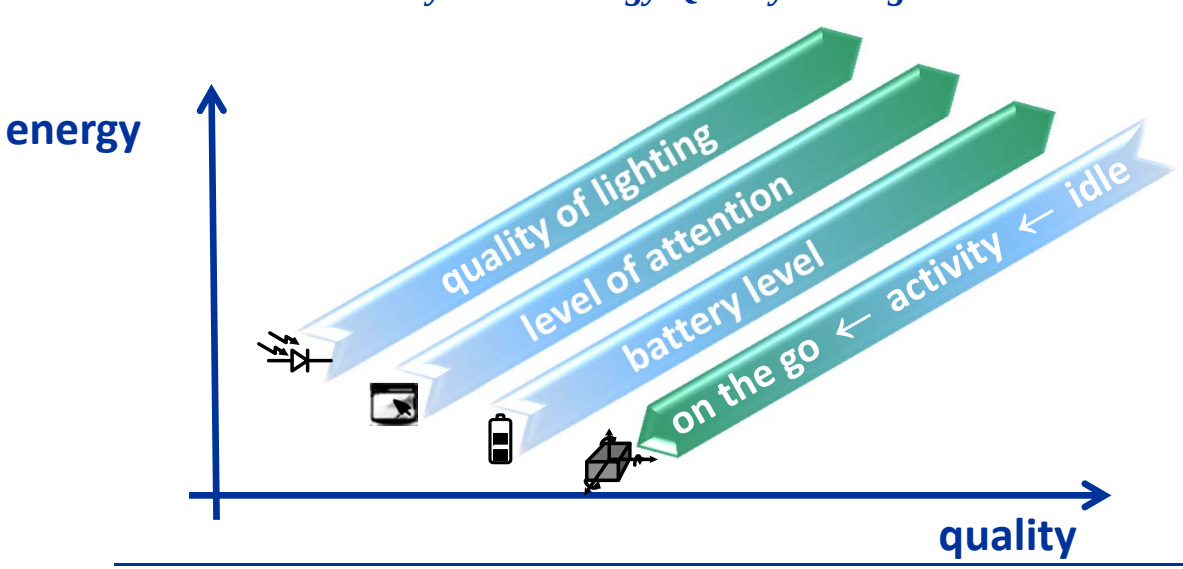
  - correction **interferes with microarchitecture** (design effort)
  - errors affect timing at boundary: **difficult SoC integration**
  - large **energy penalty** in high error rate regime (future)

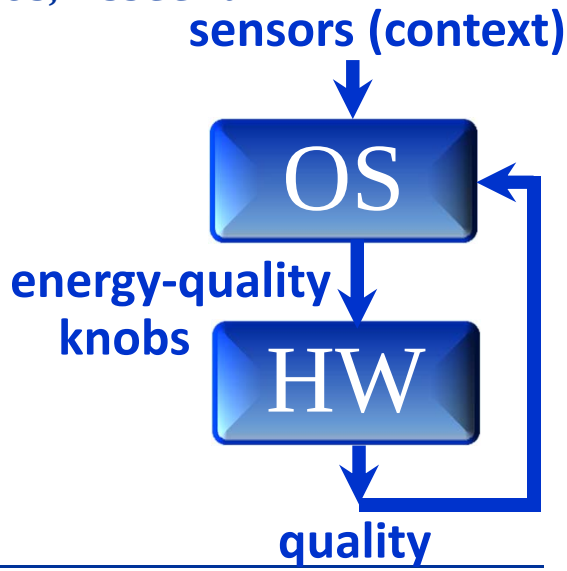
- The future of detection/correction
  - **sub-cycle** detection/correction with low overhead
    - errors detected/corrected in the same cycle
    - or, at least, errors do not have to propagate to the boundary
    - so that **errors are confined** and determine **low energy penalty**

Approximate Computing: Negative Margining

- Some apps do not need to have perfect computation
  - approximate computing (deterministic, voltage overscaling)
    - ex.: multimedia, sensor fusion
    - errors not corrected on the fly
    - **avg error rate** kept within bound (slow correction loop)
    - ex.: our first SRAM with Dynamically Adjustable Error-Quality

[FKB14] F. Frustaci, et Al., "A 32kb SRAM for Error-Free and Error-Tolerant Applications with Dynamic Energy-Quality Management in 28nm CMOS," ISSCC 2014





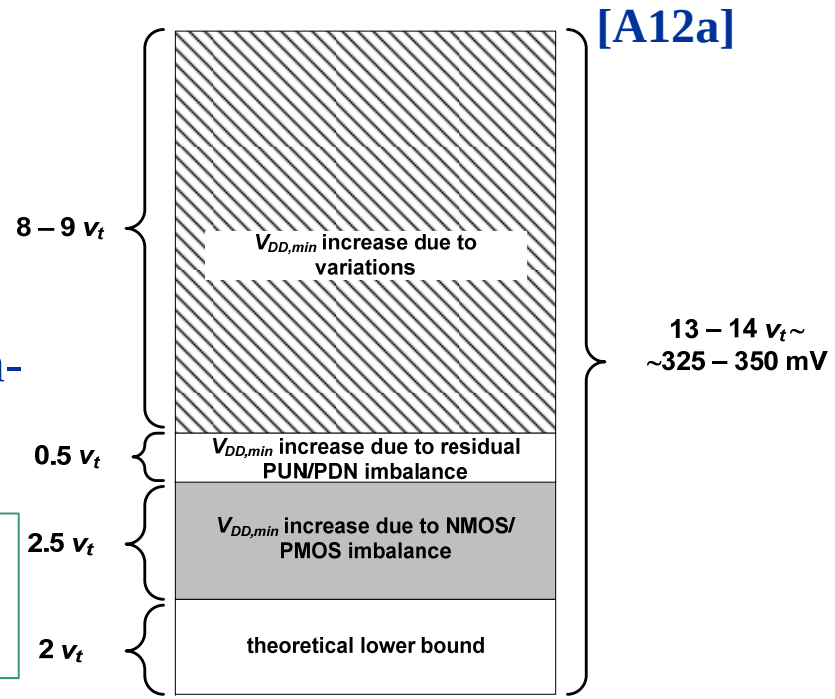
Functional Failures and $V_{DD,min}$

- Functional failures occur at very low V_{DD}
  - flip-flops/latches prone to such failures (highest $V_{DD,min}$)
    - due to multiple connected outputs (current contention due to low I_{on}/I_{off})

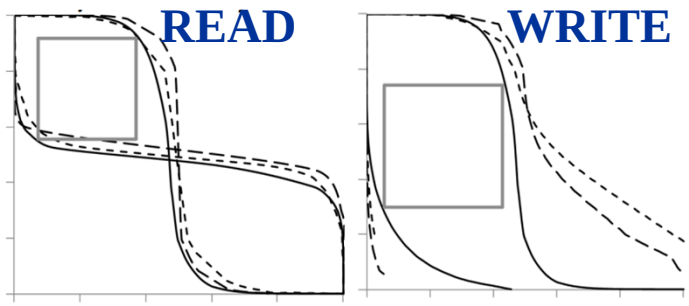
[A12a] M. Alioto, "Ultra-Low Power VLSI Circuit Design Demystified and Explained: A Tutorial," IEEE TCAS-I, Jan. 2012.

- large worst-case variations due to high number of FFs (millions)
- $V_{DD,min}$ dominated by variation-induced p/n imbalance [A12a]

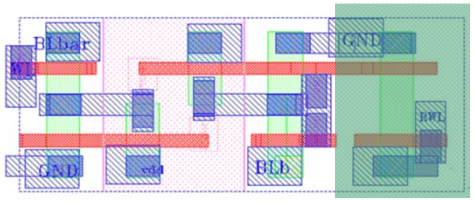
$$V_{DD,min} = n \cdot v_t \cdot \left[1 + \ln\left(\frac{2}{n}\right) + \ln(pn)\right]$$

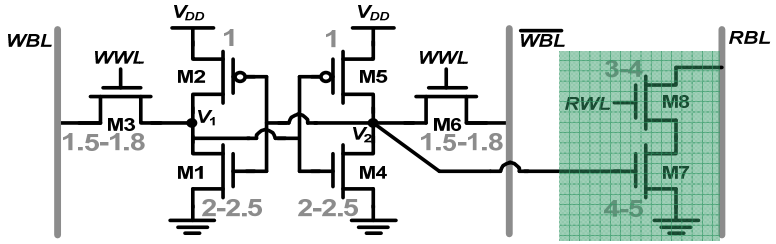


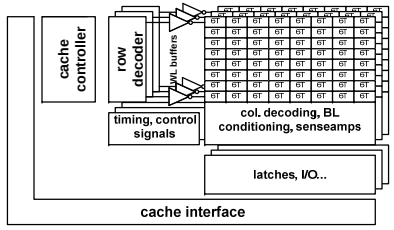
- SRAM more vulnerable than logic
  - read/write/hold margins set by strength ratio
  - no averaging across multiple cells, as opposed to logic



- reducing $V_{DD,min}$ of SRAMs at different levels
  - **within** the cell: V_{TH} adjustment, lithography-friendly layout, circuit (sizing, more robust topologies)
  - **outside** the cell: array architecture, assist techniques

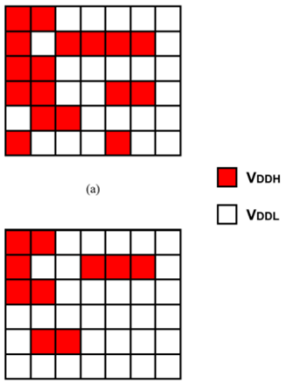






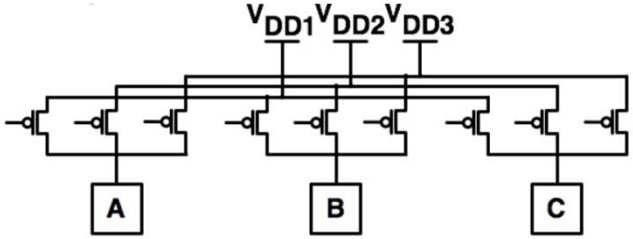
Fine-Grain Adaptation

- Multi-V_{DD} with small area/energy overhead
  - avoid level shifters altogether [MYN11]
    - small voltage domains dynamically assigned
    - no level shifters (small voltage difference)



[MYN11] A. Muramatsu, et Al. "12% Power Reduction by Within-Functional-Block Fine-Grained Adaptive Dual Supply Voltage Control (...)," ESSCIRC 2011

- Panoptic Dynamic Voltage Scaling [PDC09]
  - spatial and temporal fine granularity
  - sleep transistors (re)used to dynamically select V_{DD} (workload)
  - 34% (44%) energy saving over multi-V_{DD} (single V_{DD})

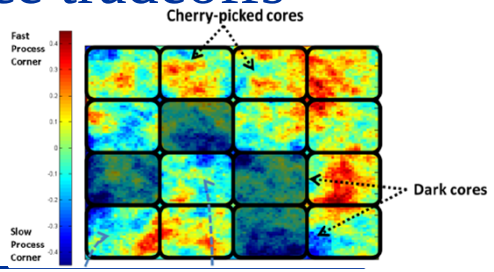


[PDC09] M. Putic, et Al., "Panoptic DVS: A Fine-Grained Dynamic Voltage Scaling Framework for Energy Scalable CMOS Design," ICCD 2009

- Selective boosting, state-retentive sleep [TRA14]
  - Graphics Execution Core
  - V_{DD} of register file/ROM **boosted** by 270 mV to improve $V_{DD,min}$
  - **adaptive clocking** reacts to first V_{DD} droop: senses and divides f_{CLK} for a fixed time to recover (margin reduction)
  - 4-20X register file **leakage reduction** in sleep mode through voltage reduction down to Data Retention Voltage of bitcells

[TRA14] C. Tokunaga, et Al. "A Graphics Execution Core in 22nm CMOS Featuring Adaptive Clocking, Selective Boosting and State-Retentive Sleep," ISSCC 2014

- Exploiting variations via **cherry picking** [RTG13]
  - different cores: different energy-performance tradeoffs
    - redundant cores, choose for max performance
    - 22% better performance at 33% dark silicon

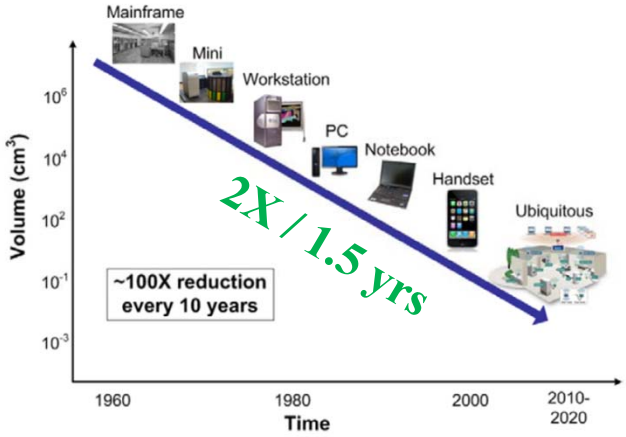


[RTG13] B. Raghunathan, et Al., "Cherry-Picking: Exploiting Process Variations in Dark-Silicon Homogeneous Chip Multi-Processors," DATE 2013

Conclusions

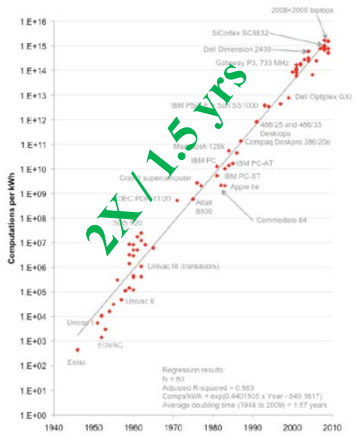
IoT Naturally Follows Historical Trends...

- **Size** is a technology driver for the IoT
  - **Bell's law**: 10-100X size reduction every 10 years
    - IoT should happen in this decade



by courtesy of D. Blaauw

- **Energy** is the bottleneck for size
  - **Koomey's law** [KBS10]: 2X / 1.6 years
    - 75X in 10 years (~4X from technology scaling)
    - rest of it must come from ckt/architecture/system
    - quicker development, more aggressive reduction: more innovation



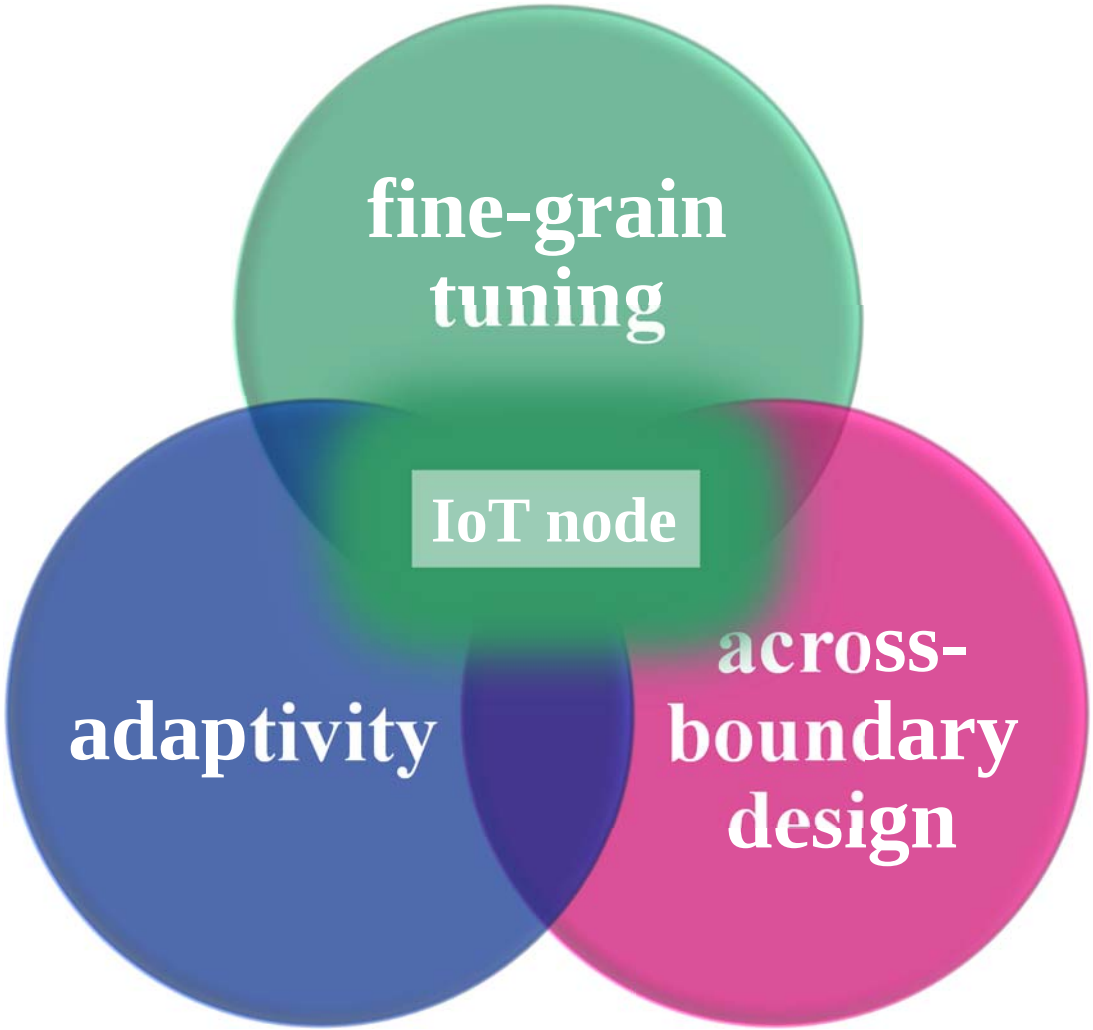
[KBS10] J. Koomey, et Al., "Implications of Historical Trends in the Electrical Efficiency of Computing" IEEE Annals of the History of Computing, March 2011

Challenges and Ideas for IoT

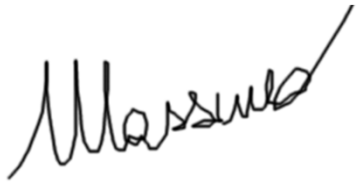
- IoT needs ultra-low voltage operation
  - energy vs power, **NT** vs **ST**
- Ultra-low voltage systems
  - tighter interaction between levels of abstraction
  - performance: lower but very sensitive to V_{DD}
    - **fine-grain** selective voltage boosting
- Issues and solutions at ULV
  - performance
    - NT: recovered architecturally (parallelism, specialized HW, wide voltage scaling
    - **across-boundary** design needed (computation vs communication)

- leakage
  - sets minimum energy point
  - stacking, power gating, multi-V_{TH} ineffective
  - fine-grain power gating and voltage domains, microarchitecture-circuit co-design, single-V_{TH} selection/body biasing
  - tight interaction of process/circuit/μarchitecture/architecture

- variations and resiliency
  - cause large cycle margin (performance/energy penalty)
  - very large at NT/ST
  - across-boundary error detection/correction (cycle-level and sub-cycle), approximate computing
  - $V_{DD,min}$ and **adaptive** techniques to reduce it


fine-grain tuning
IoT node
adaptivity
across-boundary design

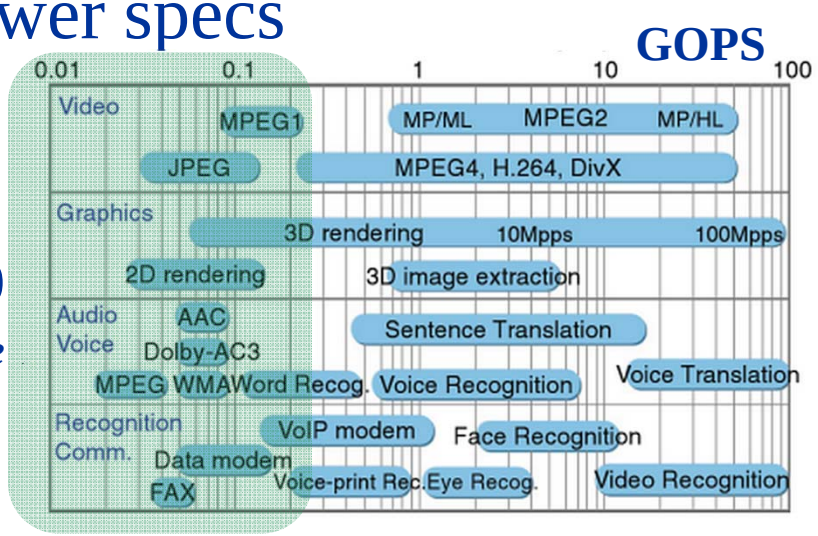
THANKS FOR YOUR ATTENTION



BACKUP SLIDES

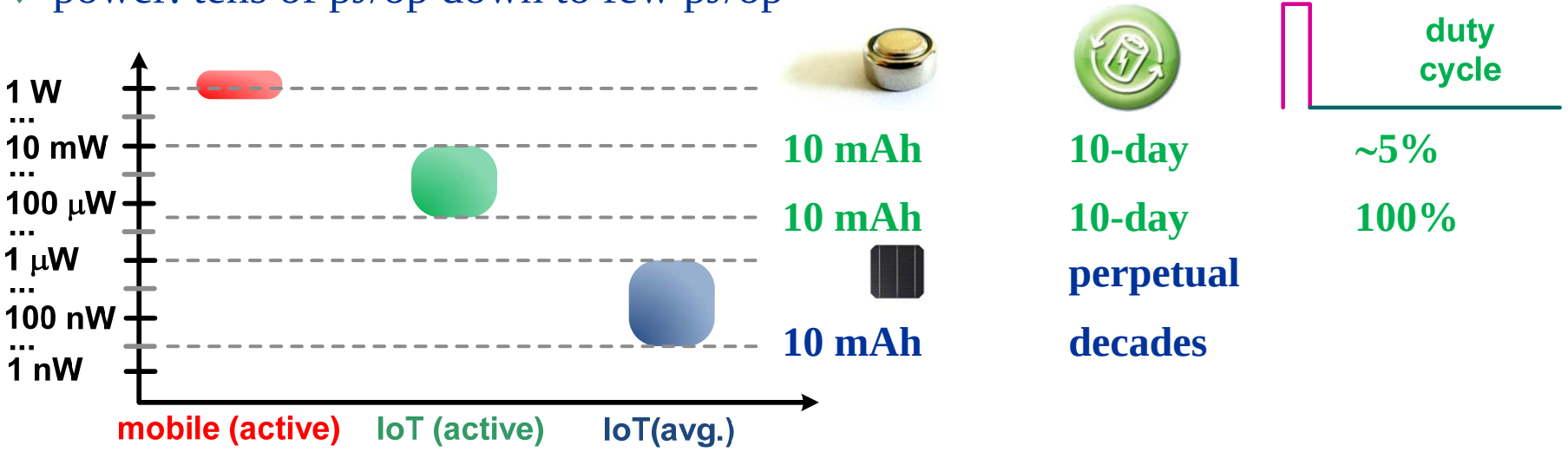
Typical performance and power specs

- throughput: tens-hundreds of MOPS down to kOPS (10X slower or more than mobile)



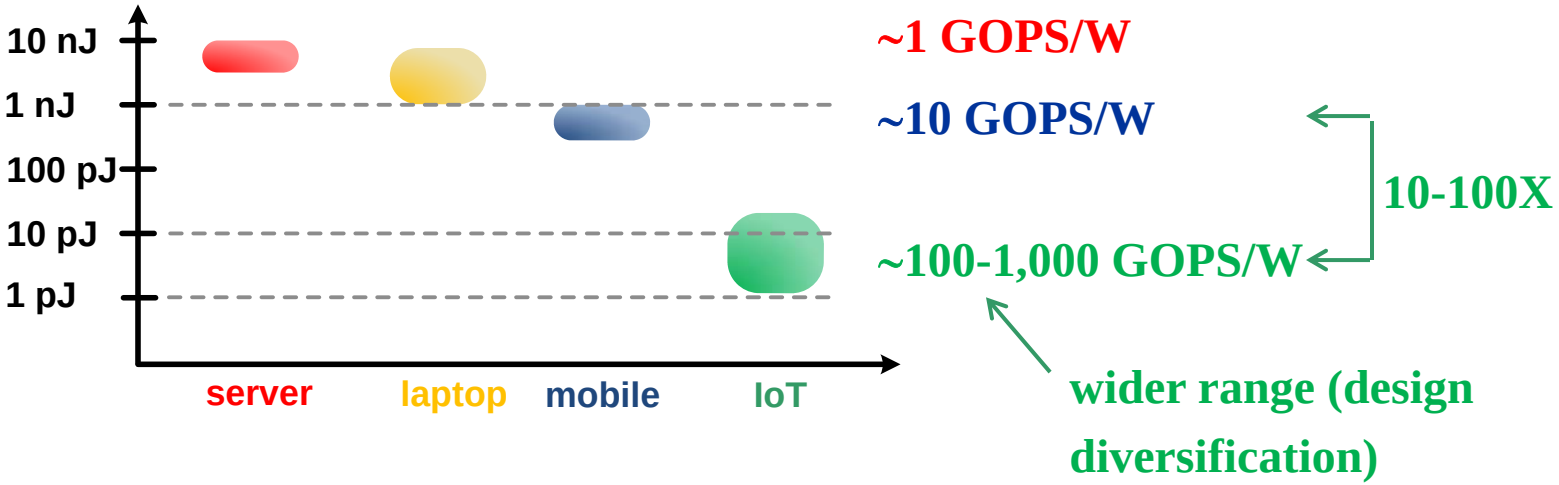
[UAK12] K. Uchiyama, et Al., Heterogeneous Multicore Technologies for Embedded Systems, Springer, 2012

- power: tens of pJ/op down to few pJ/op



Typical energy range

- energy per op: $P_{tot} = throughput \cdot E_{op} + P_{static}$



- achievable through ULV (5X), architecture tailoring (5-10X), specialized HW (10X) w.r.t. high performance
- not much gain from technology scaling...